

AD-A278 842



2

NPS-ME-93-009

NAVAL POSTGRADUATE SCHOOL
Monterey, California



**RULE BASED DESIGN OPTIMIZATION OF CRADLE STRUCTURES
USING FREQUENCY DOMAIN STRUCTURAL SYNTHESIS**

by

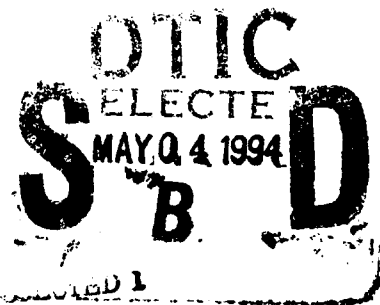
Joshua H. Gordis, Assistant Professor,
Anthony J. Healey, Professor,
Principal Investigators
and
LT Ronald E. Cook, Graduate Student

Department of Mechanical Engineering

December 31, 1993

Approved for public release; distribution is unlimited

Prepared for:
Program Executive Office - Submarines
2531 Jefferson Davis Hwy
Arlington, VA 2242-5160



DTIC QUALITY ASSURED 1

686 94-13453



94 5 03 120

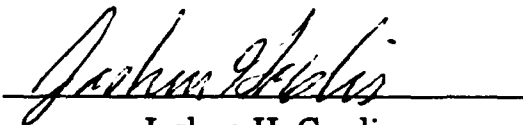
Naval Postgraduate School
Monterey, California

Real Admiral T. A. Mercer
Superintendent

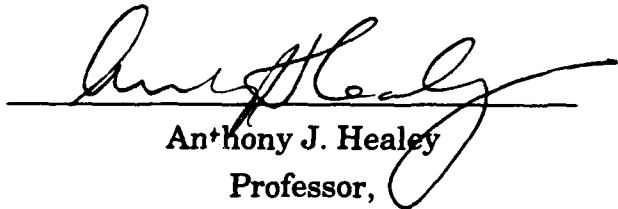
H. Shull
Provost

This report was prepared in conjunction with research sponsored by the program Executive Office - Submarines, Arlington, VA, and conducted at the Naval Postgraduate School.

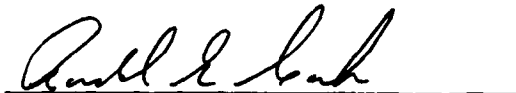
This report was prepared by:



Joshua H. Gordis
Assistant Professor,
Mechanical Engineering



Anthony J. Healey
Professor,
Mechanical Engineering



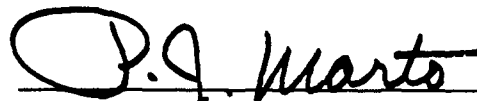
Ronald E. Cook
LT, U. S. Navy

Reviewed by:

Released by:



Matthew D. Kelleher, Chair
Dept. of Mechanical Engineering



Paul J. Marz
Dean of Research

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)		2. REPORT DATE December 31, 1993	3. REPORT TYPE AND DATES COVERED Final Report: October 1 - December 31 1993	
4. TITLE AND SUBTITLE Rule Based Design Optimization of Cradle Structures Using Frequency Domain Structural Synthesis			5. FUNDING NUMBERS N0002493WX70233	
6. AUTHOR(S) Joshua H. Gordis, Asst. Professor Anthony J. Healey, Professor, Ron Cook, LT, USN				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School, Department of Mechanical Engineering Monterey, CA 93943-5100			8. PERFORMING ORGANIZATION REPORT NUMBER NPS-ME-93-009	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Program Executive Office - Submarines 2531 Jefferson Davis Highway Arlington, VA 22242-5160			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed are those of the authors and do not reflect the official policy or position of DOD or the U. S. Government.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This work investigates the use of frequency domain structural synthesis and an expert system rule based design methodology for automating the design of the submarine machinery cradle. The expert system provides "intelligent" automated executive control of the design process. Frequency domain structural synthesis provides the means to rapidly alter the structural configuration of the cradle design and calculate dynamic response. The goal is the minimization of structural dynamic transmissibility.				
14. SUBJECT TERMS Structural Dynamics, Frequency Domain, Artificial Intelligence Optimization			15. NUMBER OF PAGES 82	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	

ABSTRACT

This research was funded under contract No. N0002493WX70233 with the objective of exploring two related ideas that offer significant advantages for the automation of the design process for submarine machinery cradle structures. As the future trends in submarine design call for more modularization in design and construction, new concepts such as the machinery cradle have to be explored from the point of view of shock isolation and acoustic radiated noise as well as manufacturability and cost. These concepts offer significant savings as modules can be assembled, tested, and installed without the need for time consuming in-situ testing and hook-up.

Two ideas with which to accelerate the design process for these structures are:

- (a) use the speed and simplicity of frequency domain structural synthesis in conjunction with conventional finite element modeling and analysis, and
- (b) develop an expert system rule base design methodology for automating the directing and coordinating of multiple design checks using a computerized implementation of the human decision making process to drive designs to a reasonable optimization. The basis of the encoding of human decision making is embodied in the AI language called 'PROLOG' - a logic programming language in which design modification decisions based on the current state of the design optimization may be encoded.

The ultimate goal of this work is to provide the basis for rapid prototyping through automated simulation-based design of submarine structures. This report represents an initial investigation into how these two separate but connected ideas may be applicable. The report is divided into Parts I and II, preceded by a General Introduction. Part I contains a discussion of Frequency Domain Structural Synthesis, a method for the rapid calculation of dynamic response, and Part II contains a discussion of the Rule Based Design Automation System.

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

ACKNOWLEDGMENT

The authors wish to express thanks to the sponsors Mr. Dan Dozier and Dr. Art Spero at the Program Executive Office - Submarines, Arlington, VA. for the financial support of this project, and also, the Naval Postgraduate School for providing the Direct Research funds necessary for our research into Intelligent Systems that gave initial impetus for the combined use of AI techniques and the structural synthesis methods for the structural design process

TABLE OF CONTENTS

ABSTRACT	1
ACKNOWLEDGMENT	2
TABLE OF CONTENTS	3
LIST OF FIGURES	4
GENERAL INTRODUCTION	5
PART 1: STRUCTURAL DYNAMIC DESIGN ANALYSIS USING FREQUENCY	
DOMAIN STRUCTURAL SYNTHESIS	7
1.0 INTRODUCTION	8
1.1 Design Changes And Structural Synthesis	9
1.2 Exact Analysis for Static and Damped Dynamic Analysis	10
1.3 Direct Synthesis of Stress and Strain	11
1.4 Structural Synthesis for Space Frames and Boolean Connectivity	11
1.5 Summary Of Capabilities Of Frequency Domain Structural Synthesis	11
2.0 THEORY	12
2.1 Generalized Frequency Response	12
2.2 Matrix Partitioning	13
2.3 Structural Modification And Indirect Substructure Coupling	15
2.4 Substructure Coupling & Constraint Imposition	16
2.5 Directed Graph S And Mapping Matrices	18
2.6 Modification And Indirect Coupling Using Mapping Matrices	19
2.7 Transformation For Indirect Coupling And Modification	19
2.8 Operative Equation For Simultaneous Direct And Indirect Synthesis	20
3.0 NUMERICAL EXAMPLES	22
Example (1) Dynamic Indirect Coupling	22
Example (2) Dynamic Direct Coupling	26
Example (3) Structural Modification (Removal of a Beam Element)	30
Example (4) Structural Modification (Addition of a Beam)	33
Example (5) Indirect Coupling with Isolators	36
Example (6) Indirect Coupling with Frequency Dependent Isolators	41
Example (7) Stress Calculation by Dynamic Indirect Coupling	44
Example (8) Dynamic Direct Coupling Using Modal Representation	50
REFERENCES	57
PART II: EXPERT SYSTEM STRUCTURE USING THE RATIONAL DESIGN	
SYSTEM (RDS) CONCEPT	58
1.0 ARTIFICIAL INTELLIGENCE AND RULE BASED SYSTEM CONCEPTS	59
1.1 General Overview	59
1.2 Artificial Intelligence, Logic, Rules and Rule Searches	59
1.3 Backward and Forward Chaining	62
1.4 The Tri-Level Architecture Concept	64
2.0 POSSIBLE RULE BASES FOR CRADLE DESIGN	66
2.1 General Overview	66
2.2 Use of 'PROLOG' Predicates for Strategic Level Rules	66
2.4 PROLOG / C Interfacing	71
3.0 EXAMPLE STRATEGIC LEVEL DESIGN	79
3.1 Introduction	79
3.2 Logical Traces - Satisfactory Design Case	79
3.3 Logical Traces - Failed Analyses	80
CONCLUSIONS	81
REFERENCES	81
INITIAL DISTRIBUTION LIST	82

LIST OF FIGURES

PART I

Figure 1 Substructure couplings and directed graphs	18
Figure 1.1. Structure analyzed using standard FE procedures.	23
Figure 1.2. Synthesis of structure	23
Figure 3.3. Plot of $H_{ee}(2,2)$ from synthesis	25
Figure 3.4. Plot $H(2,2)$ from standard FE calculation	26
Figure 2.1. Hull-cradle structure analyzed by standard FE techniques	26
Figure 2.2. Synthesis is used to directly assemble substructures	27
Figure 2.3 Plot of synthesized $H_{ee}(8,8)$	29
Figure 2.4 Plot of $H(8,8)$ from standard FE calculations.	30
Figure 3.1 Final hull-cradle configuration	31
Figure 3.2. Synthesis used to remove a beam element	31
Figure 3.3. Plot of $H_{ee}(11,11)$ as calculated using synthesis	32
Figure 3.4. Plot of $H(14,14)$ calculated using standard FE procedures	33
Figure 4.1. Hull-cradle structure analyzed by standard FE techniques.	34
Figure 4.2. Synthesis is used to add the beam element	34
Figure 4.3. Plot of synthesized FRF element $H_{ee}(8,8)$	35
Figure 4.4. Plot of $H(8,8)$ calculated using standard FE procedures.	36
Figure 5.1. Plot of FRF for the combined hull-isolator-cradle structural system.	37
Figure 5.2 Total hull-isolator-cradle system is synthesized from components.	37
Figure 5.3 Plot of $H_{ee}(8,8)$ for synthesized system	40
Figure 5.4 Plot of $H(8,8)$ from standard FE calculations.	40
Figure 6.1. Plot of FRF for the combined hull-isolator-cradle system	41
Figure 6.2. Total hull-isolator-cradle system is synthesized from components.	41
Figure 6.3. Plot of isolator damping versus frequency	42
Figure 6.4. Plot of $H_{ee}(8,8)$ calculated using the synthesis method	43
Figure 6.5. Plot of $H(8,8)$ calculated using standard FE procedures.	43
Figure 7.1 Structure analyzed for peak bending stress	44
Figure 7.2 Components of synthesized structure	45
Figure 7.3. Beam element for stress calculation	47
Figure 7.4. Beam section cut at the midpoint	47
Figure 7.5. Plot of synthesized FRF stress element $H(1,9)$	49
Figure 7.6. Plot of $H(1,6)$ calculated using standard FE procedures.	50
Figure 8.1. Structure analyzed using standard FE	50
Figure 8.2. Structures to be synthesized using modal representation	50
Figure 8.3 Plot of synthesized $H(1,3)$	54
Figure 8.4 Plot of $H(2,6)$ from standard FE calculations	54
Figure 8.5 Plot of $H(1,4)$ from standard FE calculations using modal representation ...	55
Figure 8.6 Plot of the determinant of H (plotted over reduced bandwidth)	56

PART II

Figure 1 AND / OR Diagram As The Basis For The Rule Set	63
Figure 2 Characteristics of the R D S	65
Figure 3 Rules for the Strategic Level of the RDS	69
Figure 4 PROLOG Side Foreign Files	72
Figure 5 Logical Trace of Rules - All Checks Pass	79
Figure 6 Logfile: Shock Loading Design Check Fails as Motions are Too Large	80

GENERAL INTRODUCTION

Interest in the concept of advanced machinery cradle structures has been growing recently with the paramount need to reduce costs of construction of new Naval vessels. It is a well established concept that modular structure designs lead to overall cost savings as independent modules can be fabricated and tested prior to installation in the ship, leaving less hook up time and in situ testing of components. The concept of a modular structure upon which machinery is mounted and tested prior to installation leads naturally to the use of a small number of key points upon which these modules would be connected to a submarine hull. This small number of attachment points allow for a reduced transmission path for acoustic emissions from the machinery into the submarine hull and a shock transmission path from hull to machinery that may be more readily controlled.

Such a cradle concept is now being explored by the Electric Boat Division of General Dynamics, and has lead to the need for a new mounting design of large capacity (50,000 lb.), the parameters of which may need to be adjusted to gain the most benefit considering motions, shock isolation and acoustic noise transmission between machinery and hull.

Part I of this report explores the use of Frequency Domain Structural Synthesis, a general methodology for performing rapid structural dynamic design analysis. In design, it is required to repeatedly modify and analyze the design in the search for the "optimum" design. This iterative design-analysis cycle is time consuming and expensive when dealing with large and complex structures such as the submarine machinery cradle. Frequency domain structural synthesis is ideal for the automated design analysis of submarine structures. The methodology makes possible the rapid evaluation of arbitrary design changes and the direct calculation of new performance indices, such as dynamic displacement, acceleration, stress, and when coupled with a structural acoustics analysis, radiated acoustic energy. The methodology accommodates the various additional complexities inherent in structural dynamic isolation systems, such as specialized damping treatments and frequency-dependent properties of equipment such as dynamic isolators.

Part of the problem that faces all concerned with the design of cradle structures and machinery mounts is the need to rapidly assess the impact of design changes, a requirement which is difficult because prototypes are not readily available and the effects of changes in their

design parameters not well understood. In general, too soft a mount will allow too much deflection under static loading from gravity, pitch and roll, contraction and temperature loadings, while insufficient gap in the mountings will cause acoustical shorting of the vibration isolation capabilities of the mount. Design optimization under multiple constraints is not easy to express in plain numerical terms. Therefore, it is of interest to see if the rules that human designers use in reaching satisfactory compromise can be encoded into a computer language that would coordinate the results from several disparate calculations and suggest design changes that could speed the selection process. This is the subject of Part II of this report.

The design optimization process is viewed here as a control problem with concepts borrowed from the field of Intelligent Control in Robotics. It particularly takes ideas from the control of Autonomous Underwater Vehicles where NPS is actively engaged in defining rule based motion control schemes for the coordination of complex tasks that these vehicle will be expected to accomplish in the future. We have defined an hierarchical organization of tasks for a mission control package for UUV's and it has an exact parallel for the structural design process. We define a STRATEGIC LEVEL which encodes logical decision making in arriving at conclusions as to what needs to be changed as the design progresses, a TACTICAL LEVEL, which contains the data base and characteristics of the various parts of the system as in mounts, equipment data for locations and vibration levels, weights and costs and which computations would be performed to generate new data for input to the subsequent evaluations of system response, and an EXECUTION LEVEL which contains the codes used to perform the individual analyses necessary to answer the questions and queries that the STRATEGIC LEVEL asks in forming its design change decisions. An integration of all three levels would provide the computer based designer with a valuable intelligent tool to recommend changes the would be expected to yield design improvements and could be used in an interactive way or as a stand alone system . The power of the concept is that different sets of calculations could be coordinated and managed by the TRI-LEVEL system using encoded rules that are representative of rules used in current practice.

PART I

STRUCTURAL DYNAMIC DESIGN ANALYSIS USING FREQUENCY DOMAIN STRUCTURAL SYNTHESIS

1.0 INTRODUCTION

The design of complex structural dynamic systems requires the building of detailed mathematical models with which to predict static and dynamic response. Most commonly, the finite element (FE) method is used to generate structure system matrices with which dynamic response can be calculated. While the FE method currently provides the best means of predicting response for complex structural systems, the time required to assemble the system matrices and to process them for the calculation of dynamic response can be prohibitive. Therefore, the use of the FE method for performing design analyses often precludes the performance of numerous design analyses in the search for an optimal design. This is especially true when a FE based analysis is to be used in conjunction with advanced design techniques such as optimization and artificial intelligence. The iterative process of design and subsequent analysis of the design to determine system performance is referred to as the design analysis cycle.

The traditional design-analysis cycle typically evolves as follows. A designer builds a FE model representing the current state of the design. The definition of the FE model yields system matrices. The numerical generation of the system matrices is referred to as the *assembly* phase. At this point, loads are applied and responses, static and/or dynamic, are calculated. The calculation of system response is referred to as the *solution* phase, and may involve the solution of a linear system or eigensystem. The responses are then used to calculate stresses and strains in the model. The calculation of derived quantities such as stresses and strains is referred to as the *post-processing* phase. The solution phase is the most costly in terms of time and computing resources, followed by the assembly phase, and then the post-processing phase.

Based on the acceptability of the displacements, stress, and strains so calculated, the design and hence the model may have to be changed in the interest of improving the response characteristics of the design. For example, an overly high stress may exist at a certain location in the design. The designer decides that if a particular alteration is made to the design, the stress response will be brought down to tolerable levels. Traditionally, this alteration requires a repeat of the assembly, solution, and post-processing phase of the analysis, a cost and time intensive procedure which limits the number of design re-analyses that can be accomplished, which in turn results in a final design which is less than optimal.

Therefore, with the intent of accelerating the design process and lowering the attendant costs, this section describes a design re-analysis method which allows a designer to make arbitrary changes to a model and directly calculate the new response quantities. The method replaces all three of analysis phases with a single computationally efficient calculation. The method to be described herein, generally referred to as *frequency domain structural synthesis* [1,2,3], is directed specifically at drastically reducing the time required to perform a design

analysis cycle. The method will be abbreviated simply as "structural synthesis." This capability for rapid re-analysis makes structural synthesis ideal for use in advanced automated design environments, such as in conjunction with optimization codes, and in conjunction with artificial intelligence techniques, specifically the rule based design optimization technique described in Part II of this report.

1.1 Design Changes And Structural Synthesis

With respect to a FE model, any change that a designer might make can be accomplished by a combination of the following two *synthesis* operations:

- **Addition or removal of individual finite elements:** For example, if a designer decides to increase the thickness of a plate in a hull structure, or to remove a beam from a cradle truss structure, either change can be performed by simply adding or removing element(s). The general process of adding or removing structural elements is referred to as **structural modification**.
- **Joining of components models:** For example, the coupling of two truss structures (for which individual finite element models exist) into a single truss structure model, or the coupling of a hull structural model to a cradle structural model. The general process of joining two or more structural models is referred to as **substructure coupling**.

Structural synthesis allows any number and combination of the above operations to be simultaneously performed, and hence any possible model change can be accomplished by performing the appropriate combinations of modifications and/or couplings. As just described, coupling is the joining of previously uncoupled structures and modification is the addition or removal of load paths within a structure. Each synthesis operation itself can be either *direct* or *indirect*. Indirect coupling is the joining of structures with the introduction of an intermediate or interconnecting structural element, referred to as an *interconnection impedance element*, or *impedance patch*. Indirect, or patch modification is the installation of an interconnection impedance element as a redundant load path within a given structure. Direct modification represents the application of a constraint equation to a given structural model; direct coupling is simple substructure synthesis.

An important feature of the frequency domain formulation is the arbitrary and exact model order reduction possible when performing a synthesis. A finite element method (FEM) when applied to practical problems typically generates between 10^2 to 10^5 degrees-of-freedom (DOF). The frequency domain formulation allows, as a minimum, only those DOF of interest to

be included in the analysis. This feature is in fact reason for the high computational efficiency of the method. Using one of the numerical examples presented in the following section, "Numerical Examples," the computing time required for a frequency domain synthesis can be compared with accomplishing the same analysis using standard finite element (FE) procedure. Referring to Example (6) the following count of floating point operations (FLOPS) shows the efficiency of the frequency domain method:

FEM direct assembly: Time - 25876 sec or 431.3 min.

FLOPS - 1.49×10^9

FRF synthesis: Time - 1167 sec or 19.45 min.

FLOPS - 517.2×10^6

This clearly demonstrates that synthesis by FRF is more efficient and better suited for the re-analysis of complex structures with large numbers of DOF. Moreover, the savings in time grows with increasing model size.

1.2 Exact Analysis for Static and Damped Dynamic Analysis

An important point to note is that the operations just described, and the additional features to be described below, are all performed yielding an *exact* solution; competing methods such as component mode synthesis and statistical energy analysis provide neither an exact solution nor do they provide the variety of analysis options which can be accomplished using a single, highly computationally efficient procedure.

The methodology to be described allows any linear structural element to be used as an interconnection impedance. For example, we may install a visco-elastic isolator between two points on a structure, or stiffen a structure by the addition of a plate or beam element. A cradle structural model may be attached to a hull model, and the isolation mount hardware represented by frequency-dependent visco-elastic impedances.

Operating in the frequency domain, structural synthesis is performed at each frequency of interest. This characteristic makes possible the efficient and exact treatment of frequency dependent properties, such as the shock isolator which may have complex frequency dependent elastic and damping properties. The structural synthesis methodology is unique in that it provides an exact solution for the complex damped response, again, a unique feature of the theory. The theory handles statics problems equally well.

1.3 Direct Synthesis of Stress and Strain

The theory has formulated in terms of generalized frequency response functions. These alternative frequency response functions (FRF) are defined in terms of stress and strain, and additional FRF can be defined in any response quantity of interest. The advantage here is that rather than synthesize displacement responses and then perform a post-processing step to get strains and stresses, strain and stress information can be directly synthesized. Of course, as already mentioned, the structural synthesis methodology bypasses all three phases, assembly; solution; and post-processing when performing a re-analysis for design.

1.4 Structural Synthesis for Space Frames and Boolean Connectivity

The structural synthesis theory has been tailored for the specialized problem of the design analysis of space frames, such as the machinery cradle. Space frames are comprised of beams. The geometric definition of a space frame consists primarily of defining how points in space are connected, e.g. beam "1" connects point "1" to point "2." This connectivity information can be numerically represented using boolean matrices, that is, matrices which contain 1's, -1's, and 0's. Boolean mapping matrices are well suited to the accommodation of information pertaining to *connectivity*, i.e. what is connected to what, and this information conveniently corresponds to the organization of the signs of the coupling forces and their reactions which are generated in a synthesis. The boolean mapping matrices therefore provide an "automated" way of establishing and computationally handling a sign convention for the coupling forces and reactions. The structural synthesis theory has been formulated to allow all of the above stated analysis capabilities to be performed in terms of these boolean connectivity matrices.

1.5 Summary Of Capabilities Of Frequency Domain Structural Synthesis

In summary, the advantages that the frequency domain structural synthesis methodology brings to the submarine cradle design process include:

- The theory is cast in physical coordinates, and in the frequency domain, thereby providing for an arbitrary order model reduction in conjunction with any and all of the above analysis options provided for by the theory. The theory remains exact, regardless of the order of the reduction.

- The theory makes use of frequency response functions, and therefore treats dynamic problems that can have arbitrary (linear) frequency dependent damping, and in general, any type of frequency dependency in any of the system parameters. Complex modes are easily accommodated. Static problems are also treated as the zero-frequency case.
- The theory allows for the direct synthesis of response information of any kind, a capability originally put forth in [3]. Using a generalized definition of frequency response, displacement, velocity, acceleration, stress, and strain information may be directly synthesized. Based on this generalization, the theory will be shown to be an ideal means for doing static and dynamic design re-analysis.

These capabilities will be demonstrated in the "Examples" section which follows the "Theory" section. We now present the basic theory of frequency domain structural synthesis.

2.0 THEORY

2.1 Generalized Frequency Response

A frequency response function structural model is indicated in general as

$$\{x(\Omega)\} = [H(\Omega)]\{f(\Omega)\} \quad (1)$$

Here, $\{q\}$ and $\{f\}$ are vectors of complex-valued generalized response and excitation coordinates respectively, at a specific frequency ω , and $[H]$ is an appropriately-sized frequency response function matrix, evaluated at the frequency ω . In general, an element of the frequency response function matrix is defined as,

$$H_{ij} = \partial x_i / \partial f_j \quad (2)$$

the partial derivative of the i 'th generalized response coordinate with respect to the j 'th generalized excitation coordinate. The displacement-force frequency response at a frequency ω can be found from the impedance matrix Z ,

$$H(\omega) = Z(\omega)^{-1} \quad \text{where} \quad Z(\omega) = K - \omega^2 M + j\omega C$$

and $[K]$, $[M]$, and $[C]$ are the stiffness, mass, and damping (if available) matrices which result from the finite element assembly process, and $j=\sqrt{-1}$. Of course, $[H]$ can also be found from a

vibration test. The flexibility, which is the frequency response evaluated at zero frequency, is found analytically from a non-singular stiffness matrix,

$$[H(\omega = 0)] = [K]^{-1}$$

We will employ other types of frequency response, classified by the type of coordinates involved. For example, strain-force and stress-force frequency response are defined as,

$$H_{ij}^{\epsilon} = \partial \epsilon_i / \partial f_j \quad (2a)$$

$$H_{ij}^{\sigma} = \partial \sigma_i / \partial f_j \quad (2b)$$

and the associated frequency response relations are

$$\{\sigma\} = [H^{\sigma}] \{f\} \quad (3a)$$

$$\{\epsilon\} = [H^{\epsilon}] \{f\} \quad (3b)$$

where ϵ_i and σ_i are complex-valued strains and stresses respectively at coordinate i , at a specific frequency ω , and $\{\sigma\}$ and $\{\epsilon\}$ are sets of stress and strain response coordinates, respectively.

2.2 Matrix Partitioning

Consider a structural system comprised of either a single structure for which a structural modification(s) is to be made, or two or more substructures to be coupled. The set of all physical coordinates which describe the structure(s) are denoted as coordinate set "e". The set e is comprised of two subsets. The first subset of coordinates are those at which the modifications are to be installed or substructures are to be coupled, and this set is denoted as set "c" (connection coordinates). The second subset is the complement to set c , and is comprised of all the coordinates not associated with the modifications or couplings. This coordinate set is denoted as "i" (internal coordinates). Therefore, $e = i \cup c$.

The structural system is described in the frequency domain, and at a specific frequency, by

$$\begin{Bmatrix} x_i \\ x_c \end{Bmatrix} = \begin{bmatrix} H_{ii} & H_{ic} \\ H_{ci} & H_{cc} \end{bmatrix} \begin{Bmatrix} f_i \\ f_c \end{Bmatrix} \quad (4a)$$

We will construct a transformation, referred to as the structural synthesis transformation, which operates on equation (4a) producing the synthesized system analog, which reflects the new and/or redundant load paths installed as the result of a substructure coupling and/or a structural modification.

Returning to equation (4a), we may append, for example a set of stress coordinates, then equation (4a) becomes

$$\begin{Bmatrix} \sigma \\ x_i \\ x_c \end{Bmatrix} = \begin{bmatrix} H_{\sigma i} & H_{\sigma c} \\ H_{ii} & H_{ic} \\ H_{ci} & H_{cc} \end{bmatrix} \begin{Bmatrix} f_i \\ f_c \end{Bmatrix} \quad (4b)$$

The stress coordinates will allow the direct calculation of synthesized system stress.

In general, the connection coordinates (set c) may experience both externally applied forces and coupling/modification/constraint forces (to be established through synthesis, denoted with superscript *),

$$\{f_c\} = \{f_c^{ext}\} + \{f_c^*\} \quad (5a)$$

and by definition of the subscript "i", we may have only

$$\{f_i\} = \{f_i^{ext}\} \quad (5b)$$

Introducing equations (5a,b) into equation (4b) yields the expanded equation

$$\begin{Bmatrix} \sigma \\ x_e \\ x_c \end{Bmatrix}^* = \begin{bmatrix} H_{\sigma i} & H_{\sigma c} & H_{\sigma c} \\ H_{ei} & H_{ic} & H_{ic} \\ H_{ci} & H_{cc} & H_{cc} \end{bmatrix} \begin{Bmatrix} f_i^{ext} \\ f_c^{ext} \\ f_c^* \end{Bmatrix} \quad (6)$$

where the asterisk superscript denotes a synthesized quantity due to the fact that we have introduced (symbolically as yet) the forces of synthesis, $\{f_c^*\}$. Using the set union $e = i \cup c$, we can repartition equation (6) as

$$\begin{Bmatrix} \sigma \\ x_e \\ x_c \end{Bmatrix}^* = \begin{bmatrix} H_{\sigma e} & H_{\sigma c} \\ H_{ee} & H_{ec} \\ H_{ce} & H_{cc} \end{bmatrix} \begin{Bmatrix} f_e \\ f_c \end{Bmatrix} \quad (7)$$

where $\{f_e\} = \begin{Bmatrix} f_i^{ext} \\ f_c^{ext} \end{Bmatrix}$, the vector of externally applied forces which may exist at all physical coordinates. Note that we no longer need the "*" superscript on $\{f_c\}$.

2.3 Structural Modification And Indirect Substructure Coupling

We will develop here the governing equation for structural modification and indirect substructure coupling. Structural modification is concerned with the creation of redundant load paths in a structure; indirect substructure coupling is concerned with the creation of new load paths between uncoupled structures, with an interconnecting structural element.

The theory allows for the modification to be comprised of any number and spatial distribution of linear, frequency-dependent impedances. The modifications are to be installed at the connection coordinates, i.e. coordinate set "c." The only restriction on the modifications to be made are that they be described by

$$\{f_c\} = -[K(\Omega) - \Omega^2 M(\Omega) + jC(\Omega)]\{x_c\} \quad (8a)$$

or

$$\{f_c\} = -[Z]\{x_c\} \quad (8b)$$

where the minus sign indicates that we are considering the reactions imposed by the modifications, on the host structure.

We will construct the transformation of forces based on equation (8) to be introduced into equation (7). The resulting relationship is the "modified system" version of equation (7).

The transformation which operates on equation (7) is

$$\begin{Bmatrix} f_e \\ f_c \end{Bmatrix} = \begin{bmatrix} I & 0 \\ 0 & -Z \end{bmatrix} \begin{Bmatrix} f_e \\ x_c^* \end{Bmatrix} \quad (9)$$

The transformed version of equation (7) is

$$\begin{Bmatrix} \sigma \\ x_e \\ x_c \end{Bmatrix}^* = \begin{bmatrix} H_{oe} & -H_{oc}Z \\ H_{ee} & -H_{ec}Z \\ H_{ce} & -H_{cc}Z \end{bmatrix} \begin{Bmatrix} f_e \\ x_c^* \end{Bmatrix} \quad (10)$$

The third row of equation (10) provides

$$\{x_c\}^* = [I + H_{cc}Z]^{-1}[H_{ce}]\{f_e\} \quad (11)$$

and introducing equation (11) into the upper two rows of equation (10) yields

$$\begin{bmatrix} H_{\sigma e} \\ H_{ee} \end{bmatrix}^* = \begin{bmatrix} H_{\sigma e} \\ H_{ee} \end{bmatrix} - \begin{bmatrix} H_{\sigma c} \\ H_{ec} \end{bmatrix} [Z^{-1} + H_{cc}]^{-1} [H_{ce}] \quad (12)$$

Equation (12) is the operative equation for structural modification. All terms on the right hand side are frequency response quantities for the pre-synthesis structure, and $[Z]$ describes the structural modifications to be installed. The quantities on the left hand side are frequency response values for the synthesized structure. Note that the quantity $[H_{\sigma e}]$ makes possible the direct calculation of stress due to externally applied loads for the synthesized structure.

2.4 Substructure Coupling & Constraint Imposition

We now develop the analogous theory the substructure coupling. It will be clear that the developments for substructure coupling apply to constraint imposition as well, the only difference being that coupling involves two substructures and constraint imposition involves one. We will consider here only direct coupling; interconnection impedances (indirect coupling) will be treated in the next section.

We begin again with equation (7), where the existence of two uncoupled substructures is manifest in the appropriate off-diagonal elements in each partition being zero. Coupling is to be established between pairs of coordinates in the "c" coordinate set, where the two coordinates in each pair are each from distinct substructures in a coupling, and from the same structure in the imposition of a constraint. The coupling of coordinates results in their merging into a single coordinate.

We extract the third row of equation (7),

$$\{x_c\}^* = [H_{ce}]\{f_e\} + [H_{cc}]\{f_c\} \quad (13)$$

and construct the conditions for equilibrium and compatibility to be imposed on the "c" coordinates. For the purpose of example, we will focus on a single pair of coordinates to be coupled, say x_c^A and x_c^B , where the superscripts denote that the first connection coordinate is from substructure "A" and the second from substructure "B." The equilibrium and compatibility relations associated with this pair of coordinates are

$$f_c^A + f_c^B = 0 \quad x_c^A - x_c^B = 0 \quad (14a,b)$$

Considering now all pairs of coordinates to be coupled, we can rewrite equations (14a,b) as

$$\{f_c\} = [M]\{\tilde{f}_c\} \quad (15a)$$

$$\{\tilde{x}_c\} = [M]^T \{x_c\} \quad (15b)$$

where the tilde overstrike indicates the arbitrarily selected independent subset of the "c" coordinates.

We now assemble the transformation, analogous to equation (9), which operates on equation (7), and produces the synthesized version of equation (7) describing exactly the coupled system. The transformation is assembled from equations (15a,b), and is

$$\begin{Bmatrix} f_e \\ f_c \end{Bmatrix} = \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \begin{Bmatrix} f_e \\ \tilde{f}_c \end{Bmatrix} \quad (16a)$$

$$\begin{Bmatrix} \sigma \\ x_e \\ \tilde{x}_c \end{Bmatrix}^* = \begin{bmatrix} I & 0 \\ 0 & M^T \end{bmatrix} \begin{Bmatrix} \sigma \\ x_e \\ x_c \end{Bmatrix}^* \quad (16b)$$

Operating on equation (7) these transformations produce

$$\begin{Bmatrix} \sigma \\ x_e \\ \tilde{x}_c \end{Bmatrix}^* = \begin{bmatrix} H_{\sigma e} & H_{\sigma c} M \\ H_{e e} & H_{e c} M \\ M^T H_{c e} & H_{c c} M \end{bmatrix} \begin{Bmatrix} f_e \\ \tilde{f}_c \end{Bmatrix} \quad (17)$$

Extracting the third row of equation (17) and imposing the compatibility condition between the "c" coordinates,

$$\{\tilde{x}_c\} = \{0\}$$

yields the following

$$\{\tilde{f}_c\} = -[\tilde{H}_{cc}]^{-1} [M]^T [H_{ce}] \{f_e\} \quad (18)$$

Extracting the first two rows of equation (17) and substituting equation (18) leads to the operative equation for direct substructure coupling

$$\begin{bmatrix} H_{\sigma e} \\ H_{e e} \end{bmatrix}^* = \begin{bmatrix} H_{\sigma e} \\ H_{e e} \end{bmatrix} - \begin{bmatrix} H_{\sigma c} \\ H_{e c} \end{bmatrix} [M] [\tilde{H}_{cc}]^{-1} [M]^T [H_{ce}] \quad (19)$$

where $[\tilde{H}_{cc}] = [M]^T [H_{cc}] [M]$. In equation (19), all terms in the right hand side are frequency response values calculated or measured from the uncoupled substructures. The term on the left

hand side reflects the coupled system response. Note again that the synthesis provides coupled system stress response directly. Also note that equations (15a,b) define the differential coordinate system, which is a reduced-order system.

2.5 Directed Graph S And Mapping Matrices

The use of equation (19) to perform substructure coupling requires the construction of the mapping matrices, $[M]$. As was developed in the preceding section, each column of $[M]$ represents a statement of the equilibrium and compatibility which is enforced for each pair of connection coordinates being coupled. We will now demonstrate that $[M]$ can be constructed from a graph which is drawn to represent the connectivity to be established through the synthesis.

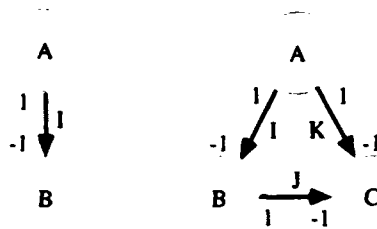


Figure 1 Substructure couplings and directed graphs

Consider the coupling depicted on the left in Figure 1. Substructure "A" is being coupled to substructure "B," through, say, a single pair of connection coordinates, x_C^A and x_C^B . The coupling of this pair of coordinates creates load path "I." To construct the mapping matrix for this connection "I," we arbitrarily assign a value of "1" to the connection coordinate of substructure "A" and a value "-1" to the connection coordinate of substructure "B." The mapping matrix for this connection is

$$[M] = \begin{bmatrix} \text{"I"} \\ 1 \\ -1 \end{bmatrix} \begin{matrix} \text{"A"} \\ \\ \text{"B"} \end{matrix} \quad (20)$$

Considering now the more complicated coupling on the right of Figure 1, and also acknowledging that in general two substructures are coupled using more than one pair of connection coordinates, we may construct the mapping matrix. Here, the connections "I", "J", and "K" consist of more than one pair of connection coordinates each; these are, in general, sets of connection coordinate pairs. The mapping matrix is

$$[M] = \begin{matrix} & \begin{matrix} \text{"I"} & \text{"J"} & \text{"K"} \end{matrix} \\ \begin{matrix} \text{"A"} \\ \text{"B"} \\ \text{"C"} \end{matrix} & \begin{bmatrix} I & 0 & I \\ -I & I & 0 \\ 0 & -I & -I \end{bmatrix} \end{matrix} \quad (21)$$

where each column contains plus/minus identity matrices whose elements correspond to the coupling to be established between each pair of connection coordinates. For example, in column 2 of the above mapping matrix, all connection coordinates associated with substructure "A" are assigned a "1" (i.e. $[I]$) and they are to be coupled to their counterparts in substructure "C" which have been assigned a "-1" (i.e. $-[I]$). The coupling of these coordinates constitutes the set of load paths denoted as "J".

The directed graphs and their boolean mapping matrices provide a means of organizing complex couplings, and also provide a framework for the computational implementation of the synthesis, i.e. equation (19). Of course, care must be exercised to insure that all matrices in equation (19) are appropriately partitioned.

2.6 Modification And Indirect Coupling Using Mapping Matrices

We now repeat the development of the operative equation of synthesis for indirect substructure coupling and structural modification. We will make use of the mapping matrices with the intent of bringing to bear the organization they provide.

Two classes of synthesis can be pursued using the mapping matrices and their directed graphs. The first is direct coupling, as discussed above. Here the mapping matrix is a boolean matrix which represents the connectivity to be established between the connection coordinates. The mapping matrix is applicable to this problem because it conveniently contains information as to the equilibrium and compatibility between the substructures.

The second class of synthesis for which the mapping matrix approach is applicable is in indirect coupling and modification. Here, the connectivity information contained in the mapping matrix must correspond to the equilibrium of the interconnecting impedance (in the case of indirect coupling), or to the equilibrium of the modification.

2.7 Transformation For Indirect Coupling And Modification

We now present the transformation matrices which operate on equation (7) and lead to the operative equation for indirect coupling and modification, using the mapping matrices.

$$\begin{Bmatrix} f_e \\ f_c \end{Bmatrix} = \begin{bmatrix} I & 0 \\ 0 & -M\tilde{Z} \end{bmatrix} \begin{Bmatrix} f_e \\ \tilde{x}_c \end{Bmatrix} \quad (22a)$$

$$\begin{Bmatrix} \sigma \\ x_e \\ \tilde{x}_c \end{Bmatrix}^* = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & M^T \end{bmatrix} \begin{Bmatrix} \sigma \\ x_e \\ x_c \end{Bmatrix}^* \quad (22b)$$

The transformed version of equation (7) is

$$\begin{Bmatrix} \sigma \\ x_e \\ \tilde{x}_c \end{Bmatrix}^* = \begin{bmatrix} H_{\sigma e} & H_{\sigma c}M\tilde{Z} \\ H_{ee} & H_{ec}M\tilde{Z} \\ M^T H_{ce} & M^T H_{cc}M\tilde{Z} \end{bmatrix} \begin{Bmatrix} f_e \\ \tilde{x}_c^* \end{Bmatrix} \quad (23)$$

The lowest row of equation (23) provides

$$\{\tilde{x}_c^*\} = [I + \tilde{H}_{cc}\tilde{Z}]^{-1} [M]^T [H_{ce}] \{f_e\} \quad (24)$$

The upper two rows of equation (23) are extracted, equation (24) is substituted, and this leads to the operative equation for indirect coupling or modification:

$$\begin{bmatrix} H_{\sigma e} \\ H_{ee} \end{bmatrix}^* = \begin{bmatrix} H_{\sigma e} \\ H_{ee} \end{bmatrix} - \begin{bmatrix} H_{\sigma c} \\ H_{ec} \end{bmatrix} [M] [\tilde{Z}^{-1} + \tilde{H}_{cc}]^{-1} [M]^T [H_{ce}] \quad (25)$$

where $[\tilde{H}_{cc}] = [M]^T [H_{cc}] [M]$.

2.8 Operative Equation For Simultaneous Direct And Indirect Synthesis

We now develop the operative equation for simultaneous indirect substructure coupling and modification, using the mapping matrices. This equation presents, in concise form, the generality of the frequency domain theory and makes clear the similarities of substructure coupling and structural modification. As already discussed, indirect substructure coupling is identical to structural modification, with respect to the derivation and application of the theory. Direct coupling and constraint imposition are also identical, and these two operations differ from the former two operations in the merging of connection coordinates versus the preservation of independence of connection coordinates.

We expand equation (7) by distinguishing between two types of connection coordinates, a subset of connection coordinates involved in a direct synthesis, $\{x_d\}$, and a subset involved in indirect synthesis, $\{x_i\}$, i.e. $c = d \cup i$.

$$\begin{Bmatrix} \sigma \\ x_e \\ x_d \\ x_i \end{Bmatrix}^* = \begin{bmatrix} H_{oe} & H_{od} & H_{oi} \\ H_{ee} & H_{ed} & H_{ei} \\ H_{de} & H_{dd} & H_{di} \\ H_{ie} & H_{id} & H_{ii} \end{bmatrix} \begin{Bmatrix} f_e \\ f_d \\ f_i \end{Bmatrix} \quad (26)$$

The transformations which operate on equation (26) are

$$\begin{Bmatrix} f_e \\ f_d \\ f_i \end{Bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & M_d & 0 \\ 0 & 0 & -M_i \tilde{Z} \end{bmatrix} \begin{Bmatrix} f_e \\ \tilde{f}_d \\ \tilde{x}_i^* \end{Bmatrix} \quad (27a)$$

$$\begin{Bmatrix} \sigma \\ x_e \\ \tilde{x}_d \\ \tilde{x}_i \end{Bmatrix}^* = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & M_d^T & 0 \\ 0 & 0 & 0 & M_i^T \end{bmatrix} \begin{Bmatrix} \sigma \\ x_e \\ x_d \\ x_i \end{Bmatrix}^* \quad (27b)$$

where $[M_d]$ and $[M_i]$ are the mapping matrices constructed for the direct and indirect connections respectively.

The transformed version of equation (26) is;

$$\begin{Bmatrix} \sigma \\ x_e \\ \tilde{x}_d \\ \tilde{x}_i \end{Bmatrix}^* = \begin{bmatrix} H_{oe} & H_{od}M_d & -H_{oi}M_i\tilde{Z} \\ H_{ee} & H_{ed}M_d & -H_{ei}M_i\tilde{Z} \\ M_d^T H_{de} & M_d^T H_{dd}M_d & -M_d^T H_{di}M_i\tilde{Z} \\ M_i^T H_{ie} & M_i^T H_{id}M_d & -M_i^T H_{ii}M_i\tilde{Z} \end{bmatrix} \begin{Bmatrix} f_e \\ \tilde{f}_d \\ \tilde{x}_i^* \end{Bmatrix} \quad (28)$$

The lower two rows provide

$$\begin{Bmatrix} \tilde{x}_d \\ \tilde{x}_i \end{Bmatrix} = \begin{bmatrix} M_d^T H_{de} \\ M_i^T H_{ie} \end{bmatrix} \{f_e\} + \begin{bmatrix} M_d^T H_{dd}M_d & -M_d^T H_{di}M_i\tilde{Z} \\ M_i^T H_{id}M_d & -M_i^T H_{ii}M_i\tilde{Z} \end{bmatrix} \begin{Bmatrix} \tilde{f}_d \\ \tilde{x}_i^* \end{Bmatrix} \quad (29)$$

Enforcing compatibility between the direct connection coordinates, i.e. $\{\tilde{x}_d\} = \{0\}$, yields

$$\begin{bmatrix} H_{oe} \\ H_{ee} \end{bmatrix}^* = \begin{bmatrix} H_{oe} \\ H_{ee} \end{bmatrix} - \begin{bmatrix} H_{od} & H_{oi} \\ H_{ed} & H_{ei} \end{bmatrix} \begin{bmatrix} M_d & 0 \\ 0 & M_i \end{bmatrix} \begin{bmatrix} M_d^T H_{dd}M_d & M_d^T H_{di}M_i \\ M_i^T H_{id}M_d & \tilde{Z}^{-1} + M_i^T H_{ii}M_i \end{bmatrix}^{-1} \begin{bmatrix} M_d & 0 \\ 0 & M_i \end{bmatrix}^T \begin{bmatrix} H_{de} \\ H_{ie} \end{bmatrix}^T \quad (30)$$

which is the operative equation for general synthesis using mapping matrices. This equation will directly synthesize displacement or stress information.

3.0 NUMERICAL EXAMPLES

The following numerical examples are provided to give a detailed explanation for each type of synthesis. The results of each example are presented graphically and are compared with the standard finite element method (FEM) solution.

Three types of damping that are addressed in the numerical examples are:

Type (1): Proportional structural damping of the form:

$$[C] = \alpha[K] + \beta[M]$$

Type (2): Proportional viscous damping of the form:

$$[C] = \alpha[K]$$

Type (3): Frequency-dependent viscous damping of the form:

$$[C] = [C_0 e^{-a\Omega}]$$

Type (1) damping is used in adding damping to a substructure, and Types (2) and (3) are used in adding damping to the isolators which are a combination of spring and dampers. The system impedance matrix Z for these damping types are:

Type (1): $[Z(\Omega)] = [K] - \Omega^2[M] + j[C]$, where $[C] = \alpha[K] + \beta[M]$.

Type (2): $[Z(\Omega)] = [K] - \Omega^2[M] + j\Omega[C]$, where $[C] = \alpha[K]$.

Type (3): $[Z(\Omega)] = [K] - \Omega^2[M] + j\Omega[C(\Omega)]$, where $[C] = [C_0 e^{-a\Omega}]$.

Example (1): Dynamic Indirect Coupling

Consider the structures shown in the following figures. The structure shown in Figure 1.1 will be directly assembled by the finite element method in order to compare a standard calculation of the frequency response with that synthesized from the substructures shown in Figure 1.2.

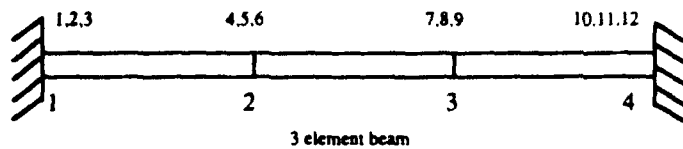


Figure 1.1. Structure analyzed using standard FE procedures.

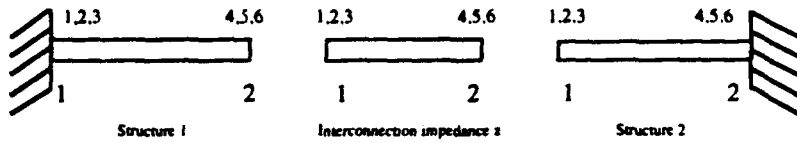


Figure 1.2. Synthesis of structure

The total structure shown in Figure 1.1 is synthesized from Structure 1 and Structure 2 through the interconnection impedance z or "new load path." For this example, the following beam parameters will be used:

Young's Modulus $E = 30.0 \times 10^6$ psi
 Area moment of inertia $I = 0.1666 \times 10^{-3}$ in⁴
 Cross-sectional area $A = 0.2$ in²
 Weight density $WTD = 0.2832$ lbf/in³
 2 percent proportional structural damping $\alpha = 0.02$
 Beam element lengths = 24 in

Proportional structural damping is applied only to structures 1 and 2, and the interconnection impedance z is undamped. The damping applied in this example was arbitrarily selected. The synthesis method is not limited to proportional structural damping, any arbitrary linear frequency dependent damping can be used. Referring to Figure 1.2, the system of structures 1 and 2 and the interconnection impedance z are synthesized in the frequency domain to yield exact results as the FEM direct assembly method. The general synthesis equation for dynamic indirect coupling is

$$[H_{ee}]^* = [H_{ee}] - [H_{ec}]([Z]^{-1} + [H_{cc}])^{-1}[H_{ce}]$$

Note again that structures 1 and 2 have proportional structural damping and the interconnection impedance z is undamped. The general procedure for performing the synthesis is as follows.

The mass and stiffness matrices $[K]$ and $[M]$ for the three substructures (including the middle beam analytically treated as an interconnection impedance) are generated using standard FEM. Since each structure is comprised of only one beam element, the elemental matrices with

boundary conditions applied are the substructure global matrices. The impedance matrix is calculated for each structure as

$$\begin{aligned}[Z_1] &= [K_1] - \Omega^2[M_1] \\ [Z_2] &= [K_2] - \Omega^2[M_2] \\ [Z_z] &= [K_z] - \Omega^2[M_z]\end{aligned}$$

Note that $[K_1]$ and $[K_2]$ are complex-valued and $[K_z]$ is real-valued. The FRF matrix $[H]$ for structures 1 and 2 is calculated by inverting the impedance matrix $[Z]$. We now have $[H_1]$, $[H_2]$ and $[Z_z]$. Referring to the general synthesis equation provided above, the matrices $[H_{cc}]$, $[H_{cc}]$, $[H_{cc}]$, and $[H_{cc}]$ are generated by assembling $[H_1]$ and $[H_2]$ by appropriate partitioning. $[H_{cc}]$ is the combination of $[H_1]$ and $[H_2]$ and is partitioned by internal and connection coordinates. Referring to Figure 1.2, after the boundary conditions are applied, coordinates 4, 5, and 6 are renumbered 1, 2, and 3 respectively for structure 1. Structure 2 is unaffected since the boundary conditions remove coordinates 4, 5, and 6 and coordinates 1, 2, and 3 remain the same. The impedance z is unchanged. $[H_{cc}]$ is partitioned in the following manner

$$[H_{cc}] = \begin{bmatrix} H(i,i) & H(i,c) \\ H(c,i) & H(c,c) \end{bmatrix}$$

where the subscript "i" represents the set of internal coordinates and "c" represents the set of connection coordinates. A more detailed representation is

$$[H_{cc}] = \begin{array}{c} \begin{array}{cc} i_1 & i_2 \\ i_1 & i_2 \\ c_1 & c_2 \end{array} \left[\begin{array}{cc|cc} H_1(i_1, i_1) & 0 & H_1(i_1, c_1) & 0 \\ 0 & H_2(i_2, i_2) & 0 & H_2(i_2, c_2) \\ \hline H_1(c_1, i_1) & 0 & H_1(c_1, c_1) & 0 \\ 0 & H_2(c_2, i_2) & 0 & H_2(c_2, c_2) \end{array} \right] \end{array}$$

In this representation "i1" denotes the internal coordinates of structure 1 and "c1" denotes the connection coordinates of structure 1. The same principle follows for "i2" and "c2" relating to structure 2. The partitioning for H_{cc} , H_{cc} , and H_{cc} are

$$[H_{cc}] = \begin{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ \begin{matrix} i_1 \\ i_2 \\ c_1 \\ c_2 \end{matrix} & \begin{bmatrix} H_1(i_1, c_1) & 0 \\ 0 & H_2(i_2, c_2) \\ H_1(c_1, c_1) & 0 \\ 0 & H_2(c_2, c_2) \end{bmatrix} \end{matrix} \quad [H_{cc}] = \begin{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ \begin{matrix} c_1 \\ c_2 \end{matrix} & \begin{bmatrix} H_1(c_1, c_1) & 0 \\ 0 & H_2(c_2, c_2) \end{bmatrix} \end{matrix}$$

$$[H_{cc}] = \begin{matrix} & \begin{matrix} i_1 & i_2 \end{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ \begin{matrix} c_1 \\ c_2 \end{matrix} & \begin{bmatrix} H_1(c_1, i_1) & 0 \\ 0 & H_2(c_2, i_2) \end{bmatrix} & \begin{bmatrix} H_1(c_1, c_1) & 0 \\ 0 & H_2(c_2, c_2) \end{bmatrix} \end{matrix}$$

In this example there are no internal coordinates. The connection coordinates for structure 1 are (1,2,3) and for structure 2 are (1,2,3). With the appropriate partitioning complete, the synthesis can now be performed. Using the indirect coupling relation

$$[H_{ee}]^* = [H_{ee}] - [H_{ec}]([Z]^{-1} + [H_{cc}])^{-1}[H_{ce}],$$

structure 1 is synthesized to structure 2 through the "new load path" z . $[H_{cc}]^*$ is the synthesized FRF relation representing the exact dynamics of the total structure. Finally the FRF relation is calculated over the frequency range 0.1 - 65 Hz and plotted in the figures which follow.

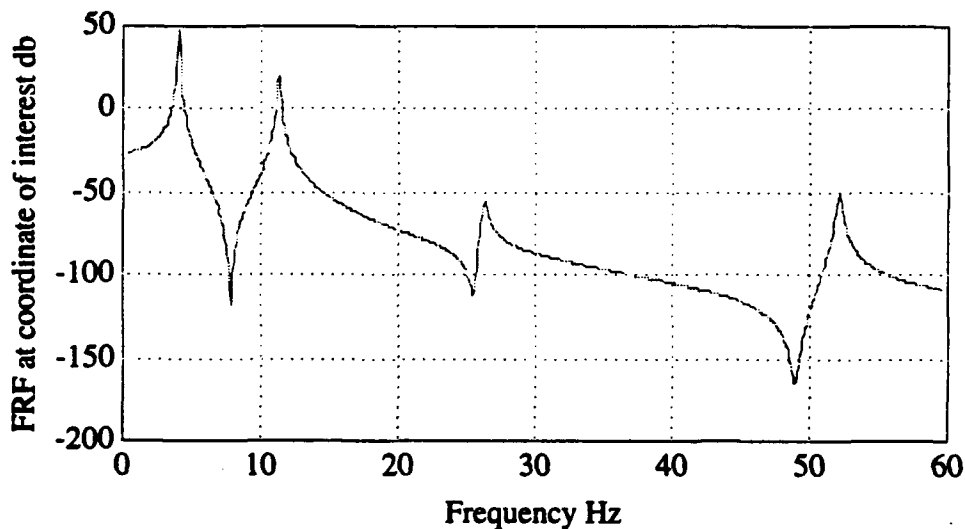


Figure 3.3. Plot of $H_{ee}(2,2)$ from synthesis

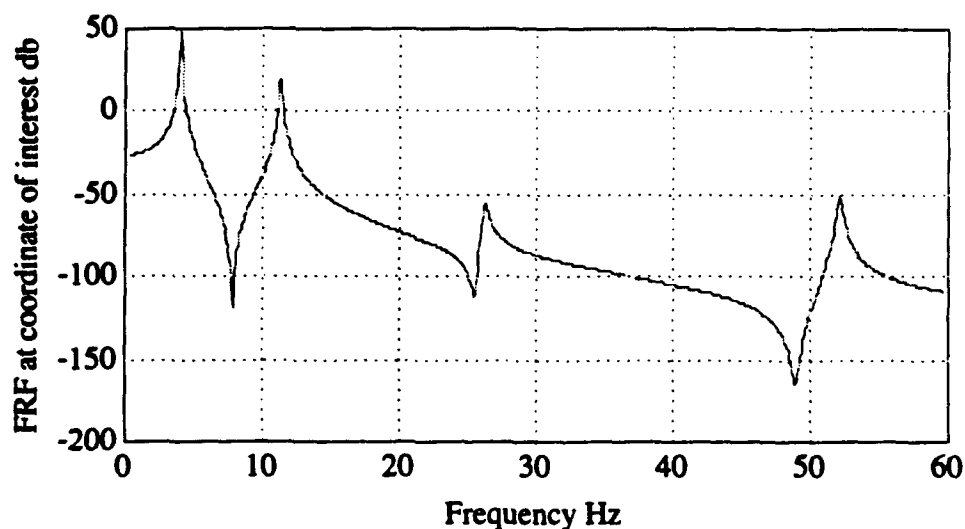


Figure 3.4. Plot H (2,2) from standard FE calculation

Figures 1.3 is a plot of the synthesized $[H_{ee}]^*$ matrix, element (2,2), and Figure 1.4 is the same FRF element calculated using the standard FE procedure. The FRF element plotted in both figures corresponds to coordinate 5 of Figure 1.1, a lateral motion coordinate. Notice both plots are identical, demonstrating that the synthesis procedure provides an exact solution for the synthesized system dynamics. The figures show the first four damped natural frequencies. The FRF plots show the magnitude of the response at coordinate 5 due to a unit excitation of varying frequency at coordinate 5.

Example (2): Dynamic Direct Coupling

Consider the following figures. The structure shown in Figure 2.1 will be directly assembled using FEM for the purpose of comparing with the results obtained by synthesizing structures 1 and 2 of figure 2.2.

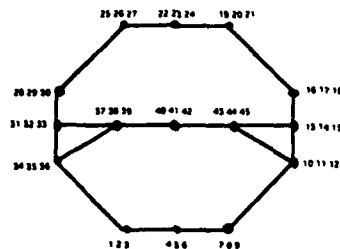


Figure 2.1. Hull-cradle structure analyzed by standard FE techniques

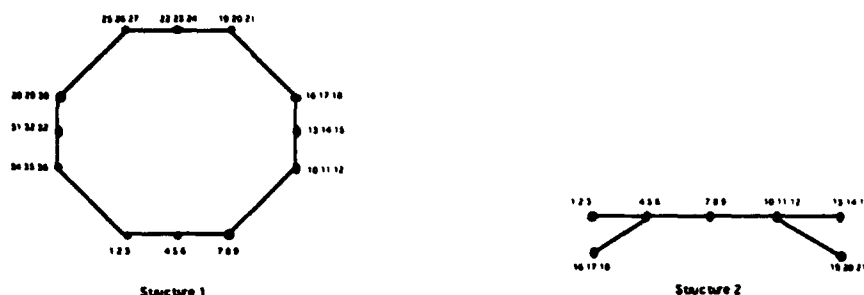


Figure 2.2. Synthesis is used to directly assemble substructures

Structure 2 will be coupled to structure 1 at coordinates 10, 11, 12, 13, 14, 15, 31, 32, 33, 34, 35, and 36. These coordinates are the connection coordinates and the remaining coordinates are internal coordinates. Coordinates 1, 2, 3, 13, 14, 15, 16, 17, 18, 19, 20, and 21 of structure 2 are connection coordinates and the remaining are internal. The following beam element data will be used:

Young's Modulus $E = 30.0 \times 10^6$ psi

Area moment of inertia $I = 0.02083$ in⁴

Cross-sectional area $A = 1$ in

Weight density $WTD = 0.2832$ lbf/in³

Proportional structural damping (1%) $\alpha = 0.01$

The proportional structural damping was arbitrarily selected and is applied to both structures. The general equation for dynamic direct coupling is

$$[H_{ee}]^* = [H_{ee}] - [H_{ec}]M([M]^T[H_{cc}]M)^{-1}[M]^T[H_{ce}]$$

In this equation, M is the boolean mapping matrix which is used to establish the connectivity between the two substructures for synthesis. The mapping matrix is determined by the connectivity i.e. what is connected to what and by imposing the equilibrium and compatibility relations associated with each pair of coordinates. We can define the mapping matrix by $\{f_c\} = [M]\{\bar{f}_c\}$. Where $\{f_c\}$ is a vector of all the connection coordinates of both structures and $\{\bar{f}_c\}$ is the arbitrarily selected independent subset of the connection coordinates relating to one of the substructures. We have selected the connection coordinates of structure 1 as the arbitrary subset of connection coordinates. The mapping matrix M is a matrix of size (24 x 12) and is depicted as:

$$[M] = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \\ & & & & & & -1 \\ & & & & & & & -1 \\ & & & & & & & & -1 \\ & & & & & & & & & -1 \\ & & & & & & & & & & -1 \\ & & & & & & & & & & & -1 \\ & & & & & & & & & & & & -1 \\ & & & & & & & & & & & & & -1 \end{bmatrix}$$

We will calculate the FRF matrix $[H]$ for both substructures. First the $[K]$ and $[M]$ matrices are generated for each substructure. $[K_1]$ and $[K_2]$ are both complex since proportional structural damping was applied to both structures; again this damping is arbitrary. $[K_1]$ and $[K_2]$ are of the form $[K] = [K + j\alpha K]$. We next form the impedance matrix for each substructure. The impedance matrix is of the form $[Z] = [K] - \Omega^2[M]$. With the impedance matrix generated for each substructure, the FRF matrix H can be calculated by inverting the impedance matrix. This process is done at each frequency of interest. These FRF matrices are required in order to couple the two structures together to form the structure in figure 2.1. Referring to the synthesis equation above, the matrices $[H_{ee}]$, $[H_{er}]$, $[H_{re}]$, and $[H_{rr}]$ are formed by combining $[H_1]$ and $[H_2]$ by appropriate partitioning. The partitioning is shown below.

$$[H_{ee}] = \begin{array}{c} i_1 \quad i_2 \quad c_1 \quad c_2 \\ \begin{array}{c} i_1 \\ i_2 \\ c_1 \\ c_2 \end{array} \left[\begin{array}{cc|cc} H_1(i_1, i_1) & [0] & H_1(i_1, c_1) & [0] \\ [0] & H_2(i_2, i_2) & [0] & H_2(i_2, c_2) \\ H_1(c_1, i_1) & [0] & H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, i_2) & [0] & H_2(c_2, c_2) \end{array} \right] \end{array}$$

$$[H_{er}] = \begin{array}{c} i_1 \quad i_2 \quad c_1 \quad c_2 \\ \begin{array}{c} c_1 \\ c_2 \end{array} \left[\begin{array}{cc|cc} H_1(c_1, i_1) & [0] & H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, i_2) & [0] & H_2(c_2, c_2) \end{array} \right] \end{array}$$

$$[H_{re}] = \begin{array}{c} c_1 \quad c_2 \\ \begin{array}{c} i_1 \\ i_2 \\ c_1 \\ c_2 \end{array} \left[\begin{array}{cc|cc} H_1(i_1, c_1) & [0] & H_1(i_1, c_1) & [0] \\ [0] & H_2(i_2, c_2) & [0] & H_2(i_2, c_2) \\ H_1(c_1, c_1) & [0] & H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, c_2) & [0] & H_2(c_2, c_2) \end{array} \right] \end{array} \quad [H_{rr}] = \begin{array}{c} c_1 \quad c_2 \\ \begin{array}{c} c_1 \\ c_2 \end{array} \left[\begin{array}{cc|cc} H_1(c_1, c_1) & [0] & H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, c_2) & [0] & H_2(c_2, c_2) \end{array} \right] \end{array}$$

Referring to figure 2.2, "i1" denotes the set of internal coordinates of structure 1 which are 1,2,3,4,5,6,7,8,9,16,17,18,19,20,21,22,23,24,25,26,27,28,29, and 30, "c1" denotes the set of connection coordinates of structure 1 which are 10,11,12,13,14,15,31,32,33,34,35, and 36, "i2" denotes the set of internal coordinates of structure 2 which are 4,5,6,7,8,9,10,11, and 12, "c2" denotes the set of connection coordinates of structure 2 which are 1,2,3,13,14,15,16,17,18,19,20, and 21. With the appropriate partitioning complete, the synthesis of structure 1 to structure 2 can be performed using the direct coupling relation

$$[H_{ee}]^* = [H_{ee}] - [H_{ec}]M([M]^T[H_{cc}]M)^{-1}[M]^T[H_{ce}].$$

$[H_{ee}]^*$ is the synthesized FRF relation which is the combination of both structures. The synthesis is done over the frequency range of interest and plotted in Figure 2.3. The frequency range for this example was 0.1 to 10.0 Hz. Figure 2.4 is the solution from standard FE calculations included for direct comparison. Both plots are identical.

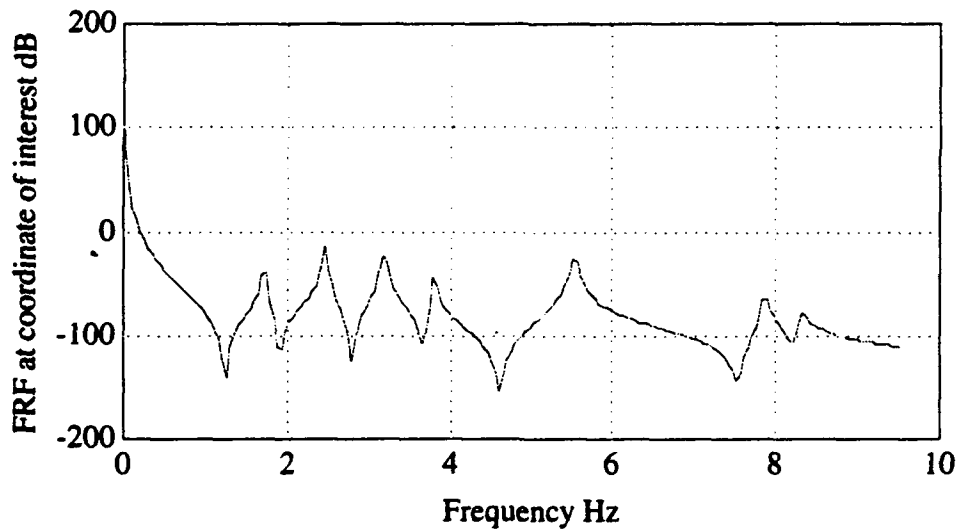


Figure 2.3 Plot of synthesized $H_{ee}(8,8)$

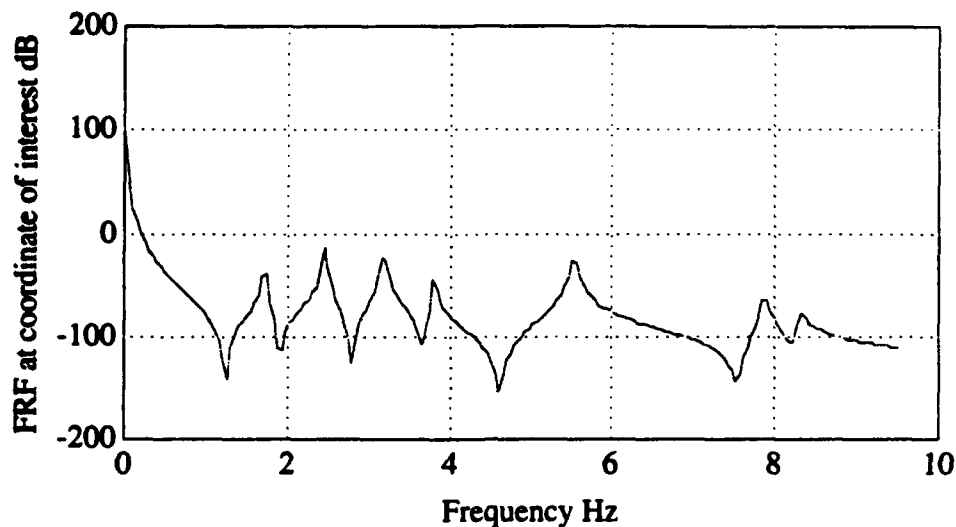


Figure 2.4 Plot of $H(8,8)$ from standard FE calculations.

Figures 2.3 and 2.4 are the plots of the FRF at element (8,8) from the synthesized and FE $[H]$ matrices. This element corresponds to the lateral motion coordinate 8 of Figure 2.1. Notice both plots are identical and both show the first seven damped natural frequencies. The plots show the magnitude of the response of unit amplitude at coordinate 8 due to a unit excitation at varying frequency at coordinate 8. As the frequency of excitation approaches the damped natural frequency, the response approaches infinity.

Example (3): Structural Modification (Removal of a Beam Element)

Consider the following figures. Figure 3.1 depicts a combined hull-cradle structure which will be directly assembled by standard FE procedures. Note that the structure in Figure 3.1 has asymmetric reinforcing trusses. The synthesis methodology will be used to arrive at the structural configuration shown on the left of Figure 3.1 by removing the beam shown on the right of Figure 3.2. The FRF calculated from the FE model (Figure 3.1) will be compared with that calculated using synthesis.

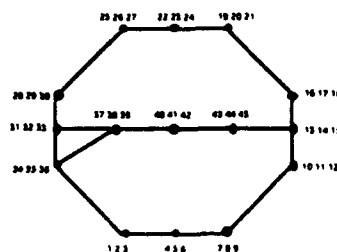


Figure 3.1 Final hull-cradle configuration

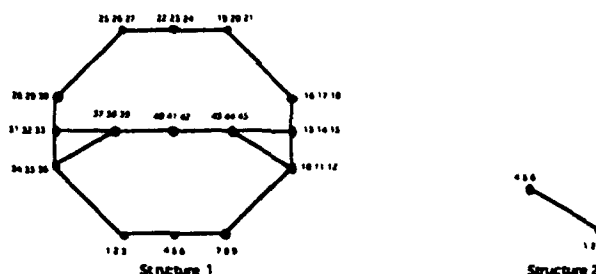


Figure 3.2. Synthesis used to remove a beam element

Referring to Figure 3.2, structure 1 will be modified by removing the beam, structure 2, located between nodal coordinates 10,11,12,43,44, and 45. The following beam element data will be used:

Young's Modulus $E = 30.0 \times 10^6$ psi

Area moment of inertia $I = 0.02083$ in⁴

Cross-sectional area $A = 1$ in

Weight density $WTD = 0.2832$ lbf/in³

1 percent proportional structural damping $\alpha = 0.01$

The proportional structural damping was arbitrarily selected and is applied to both structures. The general equation for dynamic indirect coupling/modification is

$$[H_{ee}]^* = [H_{ee}] - [H_{ec}][H_{cc}]^{-1}[H_{ce}].$$

Note that the sign in the term $([H_{cc}] - [Z]^{-1})$ is opposite from that in the original indirect coupling equation. This is because we are removing the beam element from the structure instead of synthesizing it to the structure. The first step is to generate the $[K]$ and $[M]$ matrices for structure 1 and structure 2. The $[K]$ matrices for both structures are complex since proportional damping was applied. They are of the form $[K] = [K + j\alpha K]$. Next we form the impedance matrices for each structure, $[Z] = [K] - \Omega^2[M]$. This method requires the calculation of the FRF matrix $[H]$ only for the structure to be modified, structure 1 of Figure 3.2. The impedance and the FRF matrices are calculated at the frequency of interest. Once the FRF and impedance matrices are generated, we are ready to partition the FRF matrix. The matrices $[H_{ee}]$, $[H_{ec}]$, $[H_{ce}]$, and $[H_{cc}]$ are formed by partitioning $[H_1]$. The partitioning is shown below.

$$[H_{\alpha}] = \begin{matrix} i_1 & c_1 \\ c_1 \left[\begin{array}{c|c} H_1(i_1, i_1) & H_1(i_1, c_1) \\ \hline H_1(c_1, i_1) & H_1(c_1, c_1) \end{array} \right] \end{matrix} \quad [H_{\alpha\alpha}] = \begin{matrix} i_1 & c_1 \\ c_1 [H_1(c_1, i_1) \mid H_1(c_1, c_1)] \end{matrix}$$

$$[H_{\alpha\alpha}] = \begin{matrix} c_1 \\ i_1 \left[\begin{array}{c} H_1(i_1, c_1) \\ \hline H_1(c_1, c_1) \end{array} \right] \end{matrix} \quad [H_{\alpha\alpha}] = \begin{matrix} c_1 \\ c_1 [H_1(c_1, c_1)] \end{matrix}$$

The connection coordinates for structure 1 are 10,11,12,43,44, and 45. The rest are all treated as internal coordinates. With the appropriate partitioning of $[H_1]$ completed, the removal of the beam from the structure can now be completed by using the correct form of the indirect coupling relation mentioned above. $[H_{\alpha\alpha}]$ is the synthesized FRF relation which reflects the removal of structure 2 from of structure 1. This modification is calculated over the frequency range of interest and plotted in Figure 3.3. The frequency range for this example was 0.1 to 7.0 Hz. Figure 3.4 is the solution from the standard FE procedure and is provided to allow direct comparison of the two solutions. Both plots are identical.

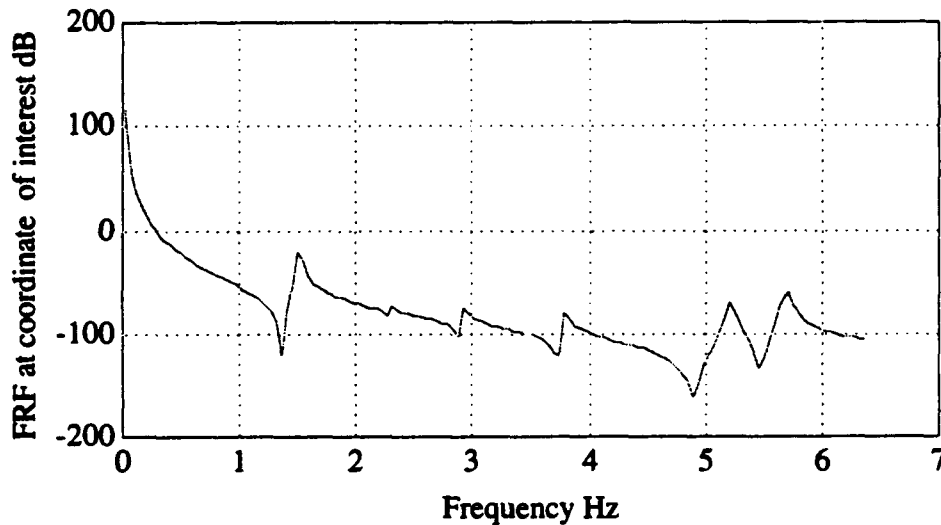


Figure 3.3. Plot of $H_{ee}(11,11)$ as calculated using synthesis

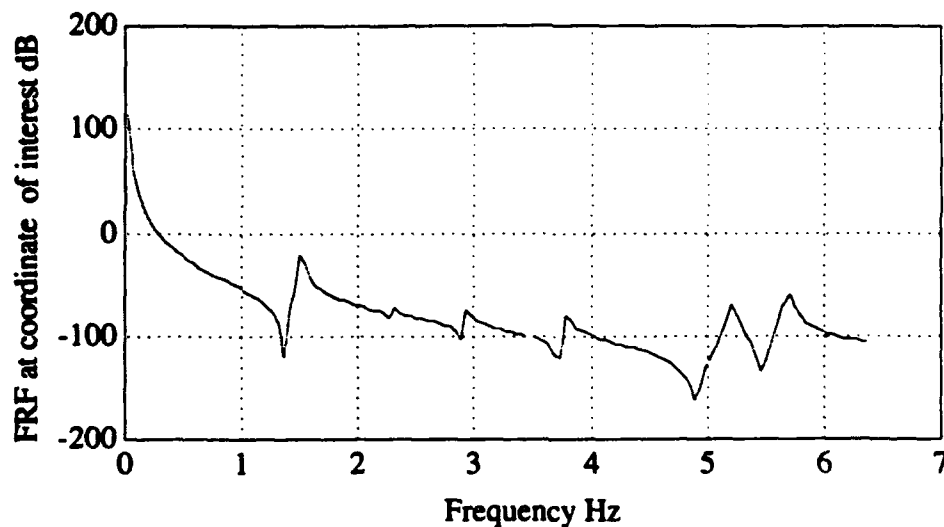


Figure 3.4. Plot of $H(14,14)$ calculated using standard FE procedures

Figures 3.3 and 3.4 are the plots of the FRF corresponding to the lateral motion coordinate 14 of Figure 3.1. A special note here is that the element (14,14) of the FRF generated by FEM is the coordinate 14, which corresponds to the element (11,11) of the FRF generated by the indirect coupling relation. The reason for this is because of the partitioning. $[H_{..}]$ is partitioned with internal coordinates first followed by the connection coordinates. Care is required here to ensure the coordinate of interest is actually being used. Notice both plots are identical and show the first six damped natural frequencies. The plots show the magnitude of the response at coordinate 14 due to a unit excitation at varying frequency at coordinate 14. As the frequency of excitation approaches the damped natural frequency, the response approaches infinity.

Example (4): Structural Modification (Addition of a Beam)

Consider the following figures. The FRF for the structure shown in Figure 4.1 will be calculated by standard FE procedures to compare with that calculated using the synthesis procedure to add the beam element, as shown in Figure 4.2.

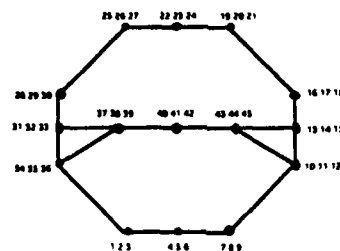


Figure 4.1. Hull-cradle structure analyzed by standard FE techniques.

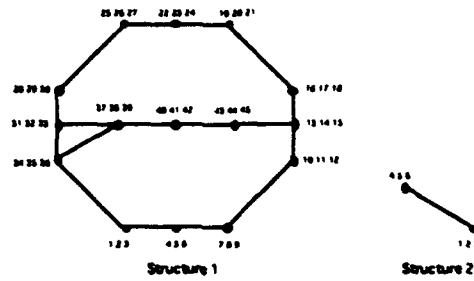


Figure 4.2. Synthesis is used to add the beam element

Referring to Figure 4.2, structure 1 will be modified by adding the beam, structure 2, at the nodal coordinates 10,11,12,43,44, and 45. The following beam element data will be used:

Young's Modulus $E = 30.0 \times 10^6$ psi
 Area moment of inertia $I = 0.02083$ in⁴
 Cross-sectional area $A = 1$ in
 Weight density $WTD = 0.2832$ lbf/in³
 Proportional structural damping (1%) $\alpha = 0.01$

The proportional structural damping was arbitrarily selected and is applied to both structures. The general equation for dynamic indirect coupling/modification is

$$[H_{ee}]^* = [H_{ee}] - [H_{ec}][H_{cc} + [Z]^{-1}]^{-1}[H_{ce}]$$

The first step is to generate the $[K]$ and $[M]$ matrices for structure 1 and structure 2. The $[K]$ matrices for both structures are complex since proportional damping was applied. They are of the form $[K] = [K + j\alpha K]$. Next, impedance matrices are formed for each structure as $[Z] = [K] - \Omega^2[M]$. This method requires the calculation of the FRF matrix $[H]$ only for the structure to be modified, structure 1 of Figure 4.2. The impedance and the FRF matrices are calculated at the frequency of interest. Once the FRF and impedance matrices are generated, partition of the FRF matrix is required. The matrices $[H_{ee}]$, $[H_{ec}]$, $[H_{cc}]$, and $[H_{ce}]$ are formed by partitioning $[H_1]$. The partitioning is shown below.

$$[H_{ee}] = \begin{matrix} & \begin{matrix} i_1 & c_1 \end{matrix} \\ \begin{matrix} i_1 \\ c_1 \end{matrix} & \left[\begin{array}{c|c} H_1(i_1, i_1) & H_1(i_1, c_1) \\ \hline H_1(c_1, i_1) & H_1(c_1, c_1) \end{array} \right] \end{matrix} \quad [H_{ce}] = c_1 [H_1(c_1, i_1) \mid H_1(c_1, c_1)]$$

$$[H_{cc}] = \begin{matrix} c_1 \\ i_1 \end{matrix} \left[\begin{matrix} H_1(i_1, c_1) \\ H_1(c_1, c_1) \end{matrix} \right] \quad [H_{cc}] = \begin{matrix} c_1 \\ c_1 \end{matrix} [H_1(c_1, c_1)]$$

The connection coordinates for structure 1 are 10,11,12,43,44, and 45. The rest are all treated as internal coordinates. With the appropriate partitioning of $[H_1]$ completed, the synthesis of the beam to the structure can now be completed by using the correct form of the indirect coupling relation mentioned above. $[H_{cc}]$ is the modified FRF relation which is the combination of structure 1 and the added element, structure 2. The synthesis is performed over the frequency range of interest and plotted in Figure 4.3. The frequency range for this example is 0.1 to 8.5 Hz. Figure 4.4 is the solution from a standard FE calculation for direct comparison of the two solutions. Both plots are identical.

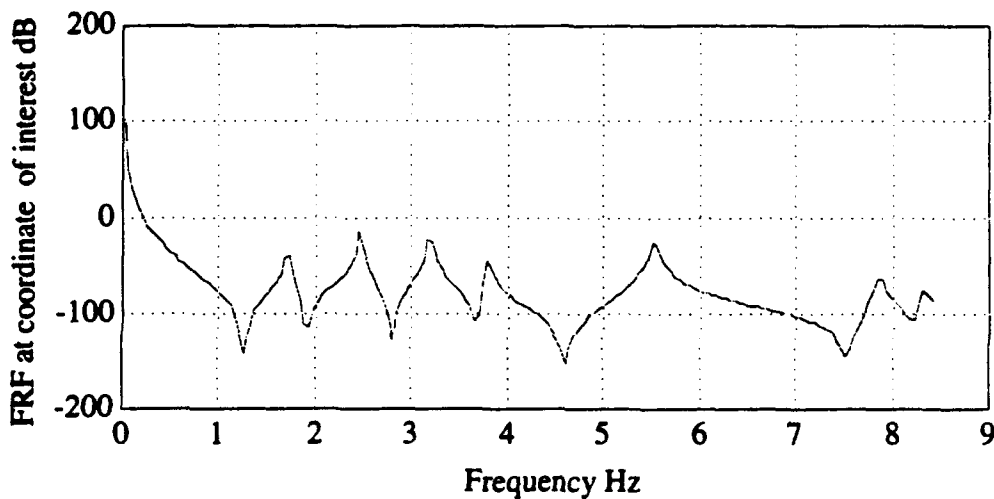


Figure 4.3. Plot of synthesized FRF element $H_{cc}(8,8)$

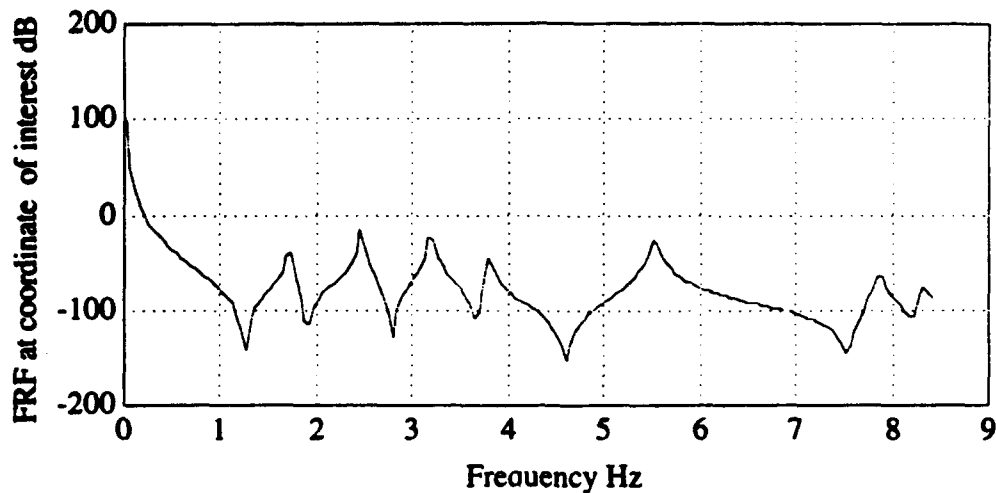


Figure 4.4. Plot of $H(8,8)$ calculated using standard FE procedures.

Figures 4.3 and 4.4 are the plots of the FRF corresponding to the lateral motion coordinate 8 of Figure 4.1. A special note here is that the element (8,8) of the FRF generated by FEM corresponds to the coordinate 8, as does the element (8,8) of the FRF generated by the indirect coupling relation. This is different from the previous example. The reason for this is because of the partitioning. $[H_{rr}]^*$ is partitioned with internal coordinates first followed by the connection coordinates. Care is required here to ensure the coordinate of interest is actually being used. Notice both plots are identical and show the first six damped natural frequencies. The plots show the magnitude of the response at coordinate 8 due to a unit excitation at varying frequency at coordinate 8. As the frequency of excitation approaches the damped natural frequency, the response approaches infinity.

Example (5): Indirect Coupling with Isolators

Consider the following figures. The FRF for the structure shown in Figure 5.1 will be calculated using standard FE procedures to compare with the FRF calculated using the synthesis method. The synthesis will combine the various components shown in Figure 5.2. In this figure, the hull model (structure 1) will be coupled to the cradle model (structure 2). Note that this example demonstrates that the synthesis procedure easily and exactly treats problems with non-proportional damping, a truly unique feature of the methodology.

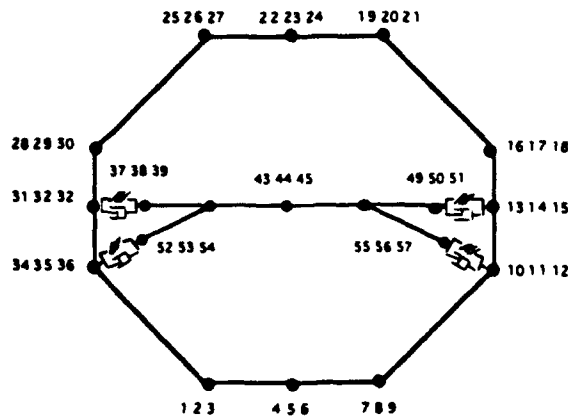


Figure 5.1. Standard FE procedures are used to calculate FRF for the combined hull-isolator-cradle structural system.

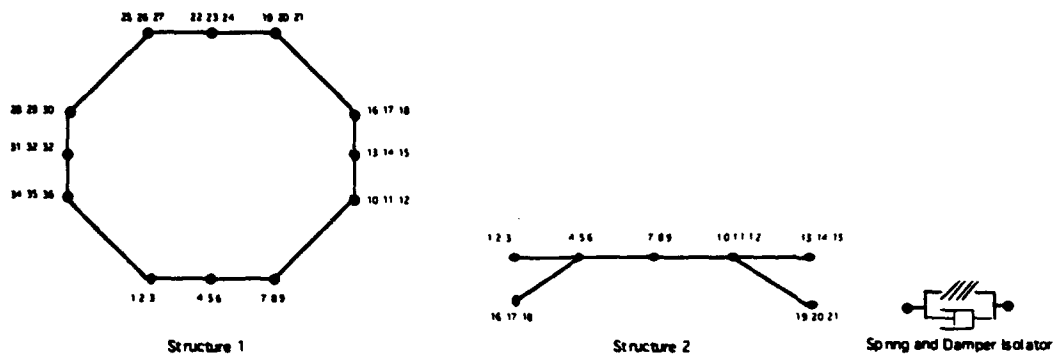


Figure 5.2 Total hull-isolator-cradle system is synthesized from components.

Referring to Figure 5.2, structure 1, structure 2, and four spring-damper isolator sets will be synthesized together to form the system in Figure 5.1. Each isolator set consists of three spring-damper isolators, one for each connection coordinate. The connection coordinates for structure 1 are 10, 11, 12, 13, 14, 15, 31, 32, 33, 34, 35, and 36. The remaining coordinates are internal coordinates. Coordinates 1, 2, 3, 13, 14, 15, 16, 17, 18, 19, 20, and 21 of structure 2 are connection coordinates and the remaining are internal. For this structural synthesis method, the spring-damper isolators are treated as a lumped system (with no physical dimensions) installed at the connection coordinates. The connection coordinates do not merge into one but are joined by way of the isolators. The following beam element data will be used:

Young's Modulus $E = 30.0 \times 10^6$ psi

Area moment of inertia $I = 0.02083$ in⁴

Cross-sectional area $A = 1$ in

Weight density $WTD = 0.2832$ lbf/in³

Proportional structural damping (2%) $\alpha = 0.02$

Proportional viscous damping (2%) $\beta = 0.02$

Isolator spring constant $k = 25 \text{ lbs/in}$

The proportional structural damping was arbitrarily selected and is applied to both structure 1 and 2 of Figure 5.2. The proportional viscous damping used for the damper in the isolator is arbitrary and is not limited to being proportional but could be any frequency dependent function. For our example the isolator is of the analytic form $[k + j\Omega\beta k]$ where $j = \sqrt{-1}$. Recalling the impedance relation $[Z(\Omega)] = [K] - \Omega^2[M] + j\Omega[C]$, $[C]$ is the proportional viscous damping, $[\beta k]$. The operative equation for indirect coupling with mapping matrices is

$$[H_{cc}]^* = [H_{cc}] - [H_{cc}]M[\tilde{Z}^{-1} + \tilde{H}_{cc}]^{-1}[M]^T[H_{cc}],$$

where $[\tilde{Z}] = [M]^T[Z][M]$ and $[\tilde{H}_{cc}] = [M]^T[H_{cc}][M]$.

Note that $[\tilde{Z}]$ reduces to $[I](k + j\Omega\beta k)$ and its size is (12 x 12). The boolean mapping matrix $[M]$ is determined the same way as explained in example two. The connection coordinates for structure 1 and structure 2 are listed above. We can define the mapping matrix by $\{f_c\} = [M]\{\tilde{f}_c\}$. Where $\{f_c\}$ is a vector of all the connection coordinates of both structures and $\{\tilde{f}_c\}$ is the arbitrarily selected independent subset of the connection coordinates relating to one of the substructures. We have selected structure 1 as the arbitrary subset of connection coordinates. The mapping matrix $[M]$ is a matrix of size (24 x 12):

$$[M] = \begin{bmatrix} 1 & & & & & & & & & & & \\ & 1 & & & & & & & & & & \\ & & 1 & & & & & & & & & \\ & & & 1 & & & & & & & & \\ & & & & 1 & & & & & & & \\ & & & & & 1 & & & & & & \\ & & & & & & 1 & & & & & \\ & & & & & & & 1 & & & & \\ & & & & & & & & 1 & & & \\ & & & & & & & & & 1 & & \\ & & & & & & & & & & 1 & \\ & & & & & & & & & & & 1 \\ & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & & & & & & 1 \\ & & & & & & & & & & & & & & & & & & & 1 \end{bmatrix}$$

The FRF matrix $[H]$ for both substructures is required. First the $[K]$ and $[M]$ matrices are generated for each substructure. $[K_1]$ and $[K_2]$ are both complex since proportional structural damping was applied to both structures, again this damping is arbitrary. $[K_1]$ and $[K_2]$ are of the form $[K] = [K + j\alpha K]$. We next form the impedance matrix for each substructure. The

impedance matrix is of the form $[Z] = [K] - \Omega^2[M]$. With the impedance matrix generated for each substructure, the FRF matrix H can be calculated by inverting the impedance matrix. This process is done at each frequency of interest. Now with the FRF matrix for each substructure calculated, we are ready to synthesize the two structures and isolators together to form the structure in Figure 5.1. Referring to the synthesis equation above, the matrices $[H_{ii}]$, $[H_{cc}]$, $[H_{ci}]$, and $[H_{ic}]$ are formed by combining $[H_1]$ and $[H_2]$ by appropriate partitioning. The partitioning is shown below.

$$[H_{ii}] = \begin{matrix} & \begin{matrix} i_1 & i_2 \end{matrix} \\ \begin{matrix} i_1 \\ i_2 \\ c_1 \\ c_2 \end{matrix} & \left[\begin{array}{cc|cc} H_1(i_1, i_1) & [0] & H_1(i_1, c_1) & [0] \\ [0] & H_2(i_2, i_2) & [0] & H_2(i_2, c_2) \\ \hline H_1(c_1, i_1) & [0] & H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, i_2) & [0] & H_2(c_2, c_2) \end{array} \right] \end{matrix}$$

$$[H_{cc}] = \begin{matrix} & \begin{matrix} i_1 & i_2 \end{matrix} \\ \begin{matrix} c_1 \\ c_2 \end{matrix} & \left[\begin{array}{cc|cc} H_1(c_1, i_1) & [0] & H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, i_2) & [0] & H_2(c_2, c_2) \end{array} \right] \end{matrix}$$

$$[H_{ci}] = \begin{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ \begin{matrix} i_1 \\ i_2 \\ c_1 \\ c_2 \end{matrix} & \left[\begin{array}{cc|cc} H_1(i_1, c_1) & [0] & & \\ [0] & H_2(i_2, c_2) & & \\ \hline H_1(c_1, c_1) & [0] & & \\ [0] & H_2(c_2, c_2) & & \end{array} \right] \end{matrix} \quad [H_{ic}] = \begin{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ \begin{matrix} c_1 \\ c_2 \end{matrix} & \left[\begin{array}{cc|cc} H_1(c_1, c_1) & [0] & & \\ [0] & H_2(c_2, c_2) & & \end{array} \right] \end{matrix}$$

Referring to figure 5.2, "i1" denotes the set of internal coordinates of structure 1 which include 1,2,3,4,5,6,7,8,9,16,17,18,19,20,21,22,23,24,25,26,27,28,29, and 30, "c1" denotes the set of connection coordinates of structure 1 which include 10,11,12,13,14,15,31,32,33,34,35, and 36, "i2" denotes the set of internal coordinates of structure 2 which include 4,5,6,7,8,9,10,11, and 12, "c2" denotes the set of connection coordinates of structure 2 which are 1,2,3,13,14,15,16,17,18,19,20, and 21. With the appropriate partitioning complete, the synthesis of structure 1 to structure 2 can be performed using the indirect coupling relation,

$$[H_{ii}]^* = [H_{ii}] - [H_{ci}] [M] [\tilde{Z}^{-1} + \tilde{H}_{cc}]^{-1} [M]^T [H_{ic}]$$

Structure 1 is synthesized to structure 2 by the isolators or load paths described by $[\bar{Z}]$. $[H_{ee}]^*$ is the synthesized FRF relation which is the combination of both structures and isolators. The synthesis is done over the frequency range of interest and plotted Figure 5.3. The frequency range for this example was 0.1 to 8.0 Hz. Figure 5.4 is the solution from standard FE calculations for direct comparison. Both plots are identical.

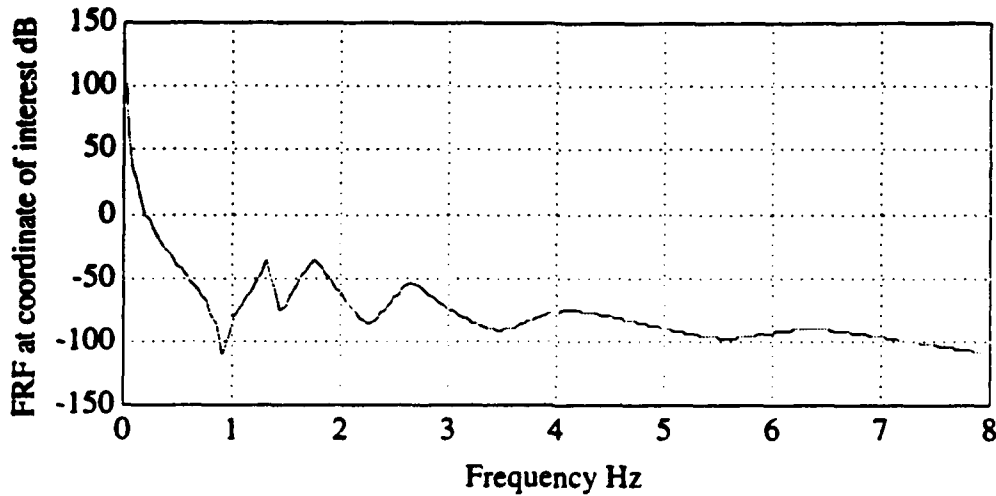


Figure 5.3 Plot of $H_{ee}(8,8)$ for synthesized system.

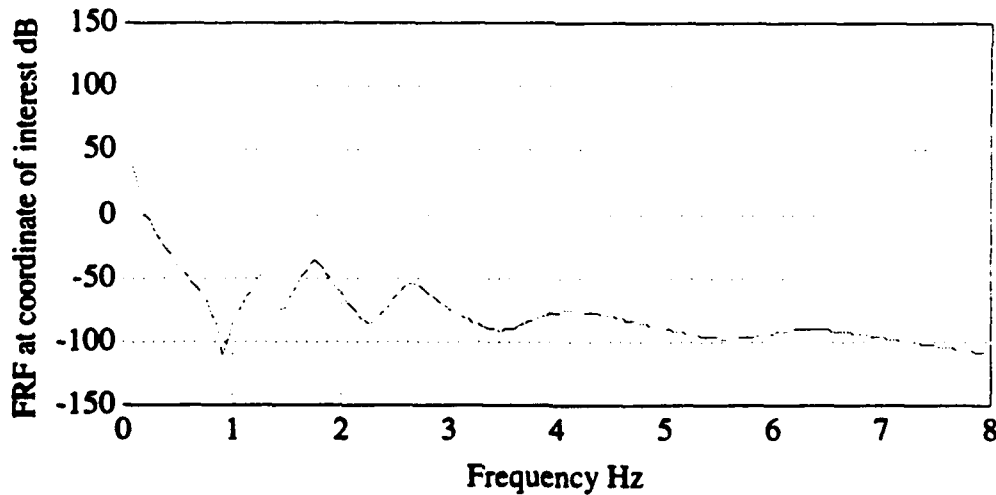


Figure 5.4 Plot of $H(8,8)$ from standard FE calculations.

Figures 5.3 and 5.4 are the plots of the FRF at element (8,8) of the synthesized $[H_{ee}]^*$ and standard FE $[H]$ matrices, respectively. This element (8,8) corresponds to the lateral motion coordinate 8 of Figure 5.1. Notice both plots are identical and show the first five damped natural

frequencies. The plots show the magnitude of the response at coordinate 8 due to a unit excitation at varying frequency at coordinate 8. As the frequency of excitation approaches the damped natural frequency, the response approaches infinity.

Example (6): Indirect Coupling with Frequency Dependent Isolators

In this example, we demonstrate the capability of synthesizing components with frequency dependent properties. Specifically, we will repeat the preceding example using isolators that have frequency dependent damping. Consider the following figures. The FRF for the structure shown in Figure 6.1 will be calculated using standard FE procedures to compare with the FRF calculated using the synthesize method. The components to be synthesized are shown in Figure 6.2.

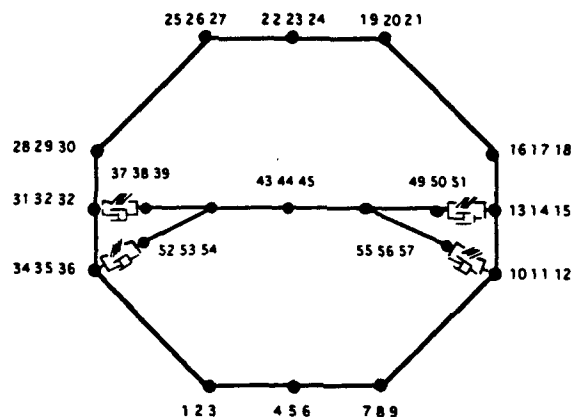


Figure 6.1. Standard FE procedures are used to calculate FRF for the combined hull-isolator-cradle structural system

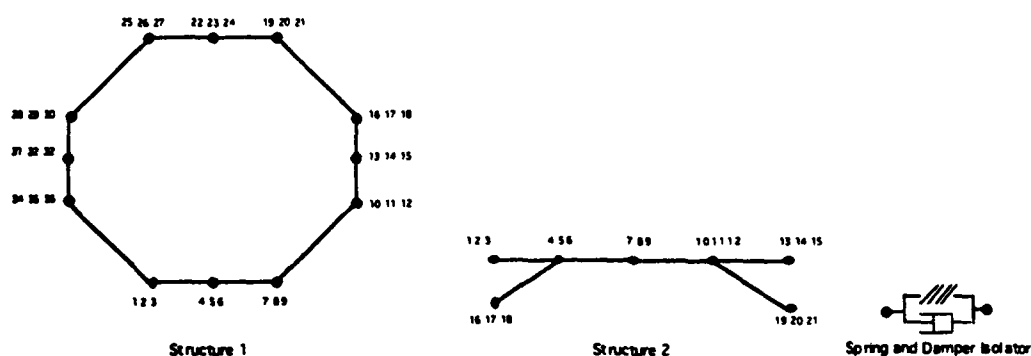


Figure 6.2. Total hull-isolator-cradle system is synthesized from components.

In Example (5), the isolators were treated as having proportional viscous damping. This example will use viscous damping which is frequency dependent. The methodology is the same as in the previous example and will not be repeated here. The discussion here will focus on the only

difference which is the damping applied to the isolator. Referring to the general impedance relation $[Z(\Omega)] = [K] - \Omega^2[M] + j\Omega[C]$, $[C]$ is now a function of Ω . The equation is now of the form $[Z(\Omega)] = [K] - \Omega^2[M] + j\Omega[C(\Omega)]$.

The damping applied to the isolator was arbitrarily selected as an exponential decay dependent on frequency. the form of the function used is

$$C = C_0 e^{-a\Omega}$$

where $C_0 = k = 25 \text{ lb} \cdot \text{s/in}$ and $a = 0.1$.

The damping function is plotted in Figure 6.3.

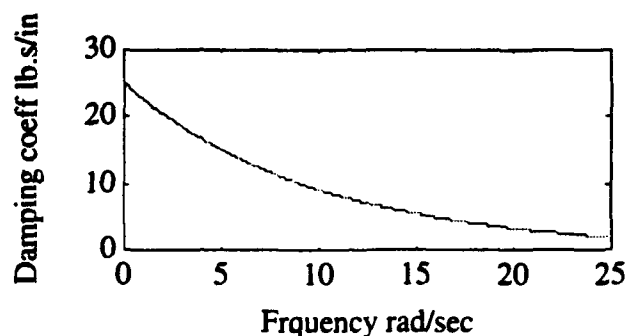


Figure 6.3. Plot of isolator damping versus frequency

The reduced impedance in this example is now of the form $[\tilde{Z}] = [I](k + j\Omega k e^{-a\Omega})$ and the size of the matrix is (12×12) .

Structure 1 is coupled to structure 2 with isolators described by $[\tilde{Z}]$. $[H_{ee}]^*$ is the synthesized FRF relation which is the combination of both structures and isolators. The synthesis is done over the frequency range of interest and plotted, Figure 6.4. The frequency range for this example was 0.1 to 8.0 Hz. Figure 6.5 is the solution from standard FE calculations provided for direct comparison. Both plots are identical.

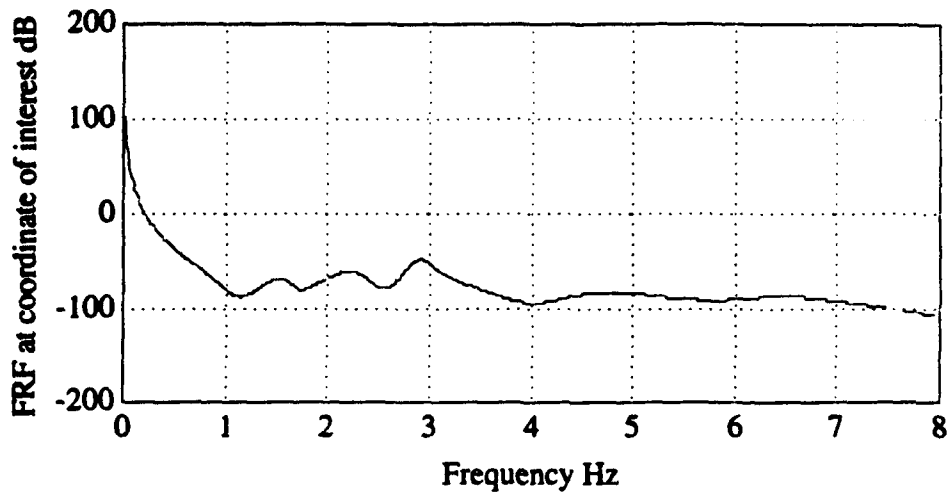


Figure 6.4. Plot of $H_{ee}(8,8)$ calculated using the synthesis method.

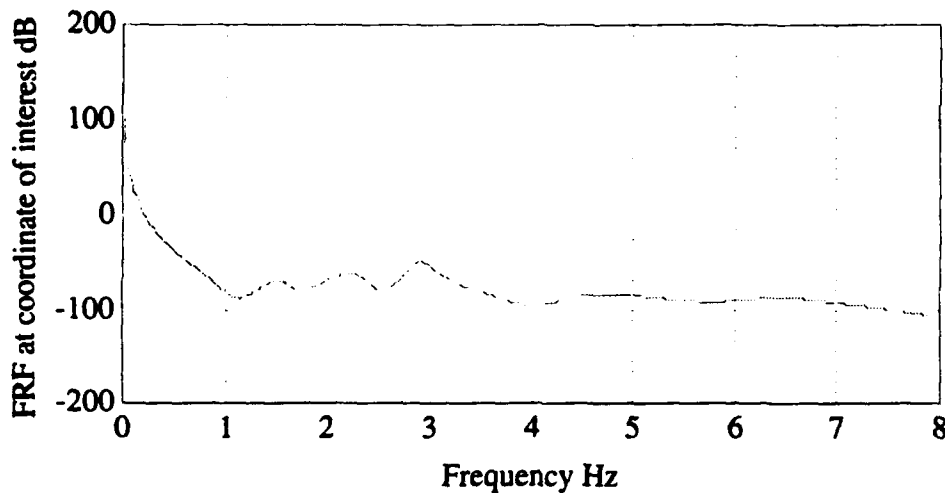


Figure 6.5. Plot of $H(8,8)$ calculated using standard FE procedures.

Figures 6.4 and 6.5 are the plots of the FRF at element (8,8) calculated using the synthesized $[H_{ee}]$ and FE $[H]$ matrices, respectively. The FRF element (8,8) corresponds to the lateral motion coordinate 8 of Figure 6.1. Notice both plots are identical and show the first five damped natural frequencies. The plots show the magnitude of the response at coordinate 8 due to a unit excitation at varying frequency at coordinate 8. As the frequency of excitation approaches the damped natural frequency, the response approaches infinity.

A comparison of the compute time required for the synthesis versus standard FE calculation. The actual computing time and the number of floating point operations (flops) for each method is provided:

FEM direct assembly: time - 25876 sec or 431.3 min.

FLOPS - 1.49×10^9

FRF synthesis: time - 1167 sec or 19.45 min.

FLOPS - 517.2×10^6

This clearly demonstrates that synthesis by FRF is more efficient and well suited for design analysis.

Example (7): Stress Calculation by Dynamic Indirect Coupling

Consider the structures shown in the following figures. The structure shown in Figure 7.1 will be directly assembled by the finite element method and the peak bending stress frequency response will be calculated in beam element #4 whose location is shown by the dashed line A--A. The same structure will be synthesized using the frequency domain method and the same stress frequency response will be calculated. The FRF results calculated by the synthesis methodology, shown in Figure 7.5, will be compared with that calculated by standard FEM, Figure 7.6. Again, the structure (specifically, its FRF) as shown in Figure 7.1 will be obtained by synthesizing structure 1 and the modification, structure 2, as shown in Figure 7.2. Note that a stress frequency response allows the direct calculation of stress due to the application of a force or moment, i.e.

$$\{x(\Omega)\} = [H_{\sigma\epsilon}(\Omega)]^* \{f(\Omega)\},$$

where the synthesized stress FRF matrix $[H_{\sigma\epsilon}(\Omega)]^*$ reflects the total structure, including any modifications or couplings.

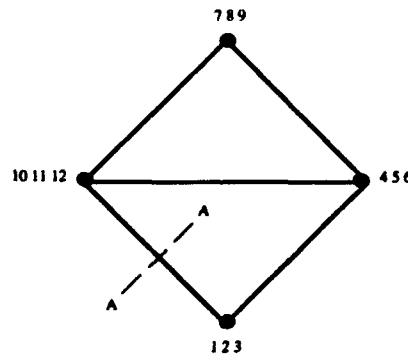


Figure 7.1 Structure analyzed for peak bending stress

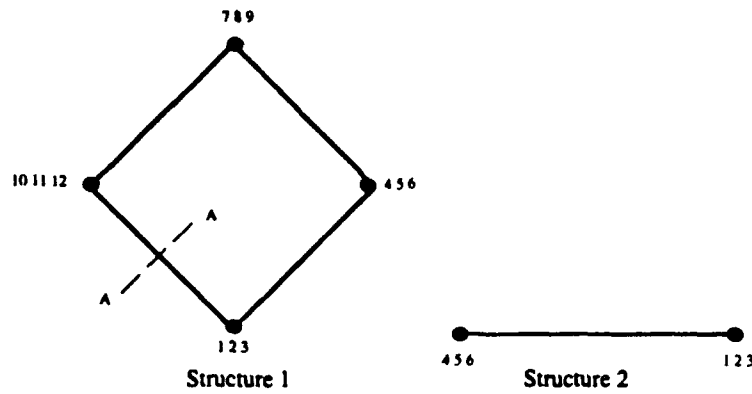


Figure 7.2 Components of synthesized structure

Referring to Figure 7.2, structure 1 will be modified by adding the beam, structure 2 at the nodal coordinates 4, 5, 6, 10, 11, and 12. The following beam element data will be used:

Young's Modulus $E = 30.0 \times 10^6$ psi

Area moment of inertia $I = 0.02083$ in⁴

Cross-sectional area $A = 1$ in

Weight density $WTD = 0.2832$ lbf/in³

Distance from beam center to outer most fiber $c = 0.05$ in

For this example, damping was not used, but the methodology is able to handle all forms of linear damping as described earlier. The general equation for synthesizing stress information by dynamic indirect coupling/modification is

$$[H_{\alpha}]^* = [H_{\alpha}] - [H_{\alpha}] [Z^{-1} + H_{cc}]^{-1} [H_{cc}].$$

This equation is the first row extracted from the relationship shown as equation (12). Note that the synthesis of stresses can be done at the same time as the synthesis of displacements. We are here demonstrating just the synthesis of stress information.

The first step is to generate the $[K]$ and $[M]$ matrices for structure 1 and the beam element shown in Figure 7.2. Next the impedance matrices are generated for each structure as $[Z(\Omega)] = [K] - \Omega^2[M]$. Since we are modifying structure 1, the FRF matrix $[H]$ is only calculated for structure 1 of Figure 7.2. The complete process as described here is performed over the frequency range of interest. There is basically two sections to this process: (1) the partitioning of the $[H]$ matrix into its required sub matrices for the general synthesis process and (2) the extraction of the information from the $[H]$ matrix and the processing of that information to calculate the stress frequency response.

Referring to Example (4), we partition $[H]$ for structure 1 in the same manner. For the synthesis of stress frequency response only, only the partitions of $[H_{cc}]$ and $[H_{ci}]$ are required. These partition are shown below.

$$[H_{cc}] = c_i \begin{bmatrix} H_i(c_i, i_i) & | & H_i(c_i, c_i) \end{bmatrix} \quad [H_{ci}] = c_i [H_i(c_i, c_i)]$$

The internal coordinates " i_i " are 1, 2, 3, 7, 8, and 9. The connection coordinates " c_i " are 4, 5, 6, 10, 11, and 12. The second part of the process requires all or part of the FRF matrix for structure 1, depending on where the external loads are applied. The connection coordinates are required for the synthesis process, as always, and internal coordinates are required if the stress frequency response which is of interest is associated with an element whose nodal coordinates are internal coordinates, i.e. they are not directly associated with the synthesis. In this example, all coordinates are used for the stress information. We apply a unit load at each coordinate using the following equation,

$$\{x\}^i = [H]\{f_i\}.$$

where i is an element of the required coordinates. This equation is interpreted as the displacements at the structural system coordinates due to the unit load at the desired coordinates of interest, which is the combination of connection coordinates and any internal coordinates desired. $\{x\}^i$ is the i 'th column of $[H]$ when using unit forces. Using the i 'th column of $[H]$, we extract the elements corresponding to the beam element that stress information is desired for, getting a partitioned form of the i 'th column. The complete reduced form of the $[H]$ matrix is $[H(bc, dc)]_{Reduced}$, where bc are the coordinates of the beam of interest and dc is the set of required coordinates we wish to keep. In our example, the set of beam coordinates, " bc " is 1, 2, 3, 10, 11, and 12 and the set of required coordinates, " dc " is 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12. $[H]_{Reduced}$ is a matrix of size (6 x 12). Each column of $[H]_{Reduced}$ is in the global coordinate system and needs to be transformed to local coordinates by using the following relation

$$\{H^i\}_{Reduced_{local}} = [T]\{H^i\}_{Reduced_{global}}$$

where the transformation matrix $[T]$ is of the form

$$[T] = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 & 0 & 0 & 0 \\ -\sin \alpha & \cos \alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & 0 & 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Now with the H transformed to the local coordinate system, we can get the nodal forces by the following relation

$$\{f^i\} = [k_{el}] \{H^i\}_{\text{Reduced}_{\text{local}}}$$

where $[k_{el}]$ is the elemental stiffness matrix in the local coordinates system. We are ready to solve for the FRF stress, first combining all the column vectors $\{f^i\}$ into a nodal force matrix $[F]$ and then multiplying it by the moment equation to solve for the FRF peak internal bending stress of the beam element. The FRF stress equation is

$$\{H_\sigma\} = \frac{c}{I} \{M_{eq}\} [F]$$

The $\{H_\sigma\}$ is a row vector size (1 x 12), since we used all the coordinates as our set of desired coordinates. Noting that $\{M_{eq}\}$ is determined from equilibrium for the element in question, and here, we will calculate the internal bending moment. The derivation is shown below.

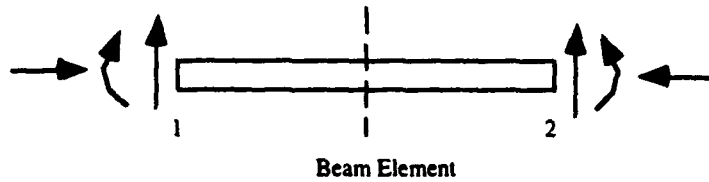


Figure 7.3. Beam element for stress calculation

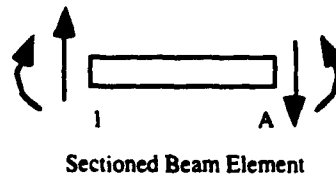


Figure 7.4. Beam section cut at the midpoint

Consider the beam element in Figure 7.3, The moment is to be calculated at the midpoint of the beam. First the beam is cut , Figure 7.4 and the moment at A is solved for:

$$M_A = M_1 + V_1 \frac{l}{2}$$

The moment equation in vector form is in the following form:

$$\{M_{eq}\} = \{M_A\} = \{0 \quad l/2 \quad 1 \quad 0 \quad 0 \quad 0\}.$$

Noting that the nodal force of the beam element is

$$\{F^i\} = \begin{Bmatrix} A_1 \\ V_1 \\ M_1 \\ A_2 \\ V_2 \\ M_2 \end{Bmatrix}.$$

where "A" indicates an axial force, "V" a shear force, and "M," a moment. The internal bending stress frequency response component is determined by the following relation,

$$H_\sigma^i = \frac{c}{I} \{0 \quad l/2 \quad 1 \quad 0 \quad 0 \quad 0\} \begin{Bmatrix} A_1 \\ V_1 \\ M_1 \\ A_2 \\ V_2 \\ M_2 \end{Bmatrix}$$

H_σ^i is evaluated over all the chosen required coordinates to form $\{H_\sigma\}$, which in our example is a row vector size (1 x 12). If more then one beam element is used for stress calculations then $[H_\sigma]$ could be of the size: (number of beam elements) x (number of desired coordinates). With $[H_\sigma]$ generated, we can now partition it into the required sub-partitions for synthesis. The partitions required are $[H_\alpha]$ and $[H_\sigma]$ and are shown below.

$$[H_\alpha] = \begin{matrix} & c1 \\ nb & [H_\sigma(nb, c1)] \end{matrix} \quad [H_\sigma] = \begin{matrix} & i1 & c1 \\ nb & [H_\sigma(nb, i1) \quad H_\sigma(nb, c1)] \end{matrix}$$

The beam element set is indicated by "nb". In this example "nb" could have been 1, 2, 3, and 4 since there are four beam elements in the substructure to be modified. Beam element 4 was chosen as the beam to calculate FRF stress information so "nb" is 4 and the sizes of the matrices are (1 x 6) and (1 x 12) respectively. To get the stress information for beam five, the synthesis method of direct coupling must be used. With the appropriate partitioning completed the synthesis can now be performed. $[H_\alpha]$ is the modified FRF stress relation which is the combination of structure 1 and the added element structure 2 of Figure 7.2. The synthesis is performed over the frequency range of interest and plotted in figure 7.5. The frequency range for this example is 0.3 to 102 Hz. Figure 7.6 is the solution from the standard FE calculation for direct comparison of the two solutions. Both plots are identical.

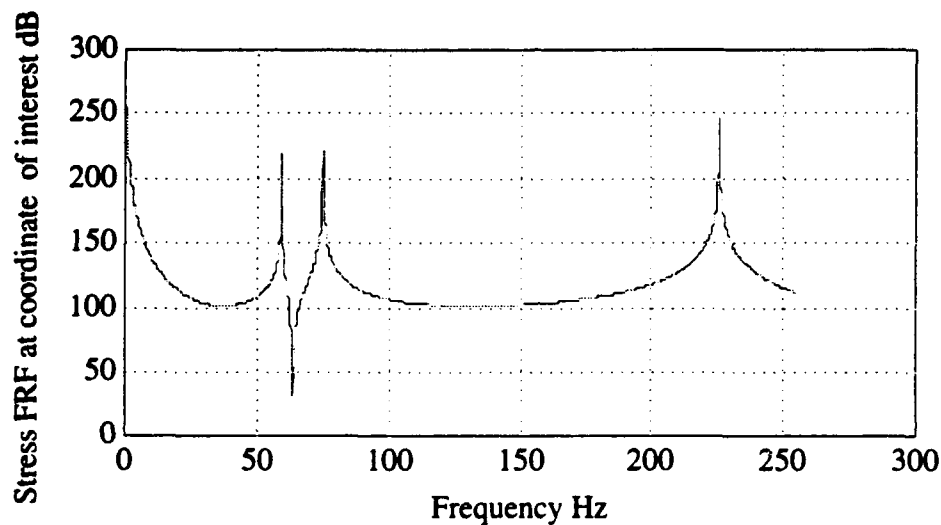


Figure 7.5. Plot of synthesized FRF stress element $H_\alpha(1,9)$

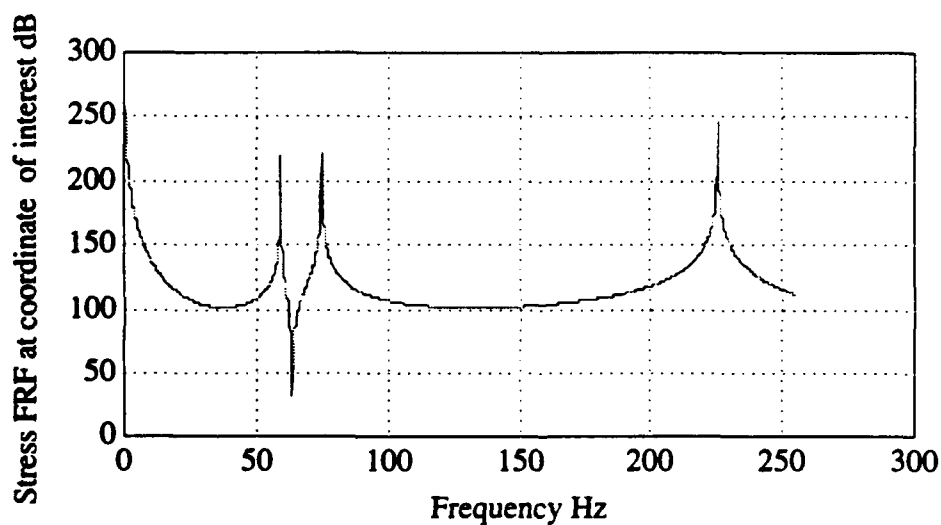


Figure 7.6. Plot of $H_o(1,6)$ calculated using standard FE procedures.

Figures 7.5 and 7.6 are the plots of the FRF stress corresponding to beam element four of Figure 7.1. These plots represent the stress amplitude in beam element four due to a unit force applied at coordinate 6. Both plots are identical. Note that the frequency of peak response is slightly lower than the undamped natural frequency.

Example (8): Dynamic Direct Coupling Using Modal Representation of FRF

Consider the structures shown in the following figures. The structure shown in Figure 8.1 will be directly assembled by the finite element method in order to compare the frequency response calculated by standard FE methods and the solution obtained by synthesizing structure 1 and structure 2 as shown in Figure 8.2. This example will show three results, the first being the solution from synthesis using the modal representation of the frequency response, the second being direct assembly using FE and the modal representation and thirdly, direct assembly using FE where frequency response is calculated by the inverse of the impedance matrix.

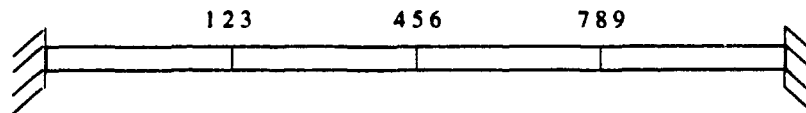


Figure 8.1. Structure analyzed using standard FE

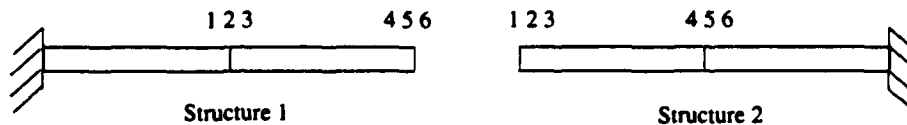


Figure 8.2. Structures to be synthesized using modal representation

Referring to Figure 8.2, structures 1 and 2 will be synthesized by direct coupling using connection coordinates 4, 5, and 6 of structure 1 and connection coordinates 1, 2, and 3 of structure 2. One internal coordinate will be kept in this synthesis process to show that the frequency response for a specific coordinate can be synthesized using just the connection coordinates and any internal coordinates that might be of interest. This example will use the internal coordinate "2" of structure 1 as the coordinate of interest. The information desired in this example is the frequency response at coordinate 2 due to a unit harmonic load at coordinate 6. Note that when structure 1 and structure 2 are synthesized, the coordinate numbering becomes the same as depicted in Figure 8.1. The following beam element data was used:

Young's Modulus $E = 30.0 \times 10^6$ psi

Area moment of inertia $I = 0.02083$ in⁴

Cross-sectional area $A = 1$ in

Weight density $WTD = 0.2832$ lbf/in³

Structural proportional damping was not used, but the methodology will handle all forms of damping discussed earlier. The frequency response matrix $[H]$ can be generated by two methods. The first is by the relationship

$$[H(\Omega)] = [Z(\Omega)]^{-1}$$

where $[Z(\Omega)] = [K] - \Omega^2[M] + j\Omega[C]$.

The second method is by matrix modal representation. The relationship is

$$[H(\Omega)] = [\Phi] \begin{bmatrix} \frac{1}{\omega_1^2 - \Omega^2} \\ \vdots \\ \frac{1}{\omega_n^2 - \Omega^2} \end{bmatrix} [\Phi]^T$$

where $[\Phi]$ is the set of eigenvectors or mode shapes, and the middle term is the diagonal matrix of the natural frequencies or eigenvalues less the frequency of interest. This relationship can also be expressed in terms of individual elements of the frequency response function by the following relationship

$$H_{ij} = \sum_{r=1}^{n \text{ modes}} \frac{\phi_i^r \phi_j^r}{\omega_r^2 - \Omega^2},$$

which allows the calculation of specific information of interest without having to generate the complete FRF.

The general synthesis equation for dynamic direct coupling is

$$[H_{\alpha\alpha}]^* = [H_{\alpha\alpha}] - [H_{\alpha\alpha}][M]([M]^T[H_{\alpha\alpha}][M])^{-1}[M]^T[H_{\alpha\alpha}].$$

This equation is written in terms of all the coordinates. This example concerns the synthesis of a single FRF matrix element involving one coordinate. In general, the synthesis process requires FRF information for all connection coordinates, and FRF information for any internal coordinates of interest. Rewriting the general equation for this specific example, the equation becomes

$$[H_{2c}]^* = [H_{2c}] - [H_{2c}][M]([M]^T[H_{cc}][M])^{-1}[M]^T[H_{cc}]$$

where the subscript "2" signifies the internal coordinate "2" of structure 1 and the subscript "c" signifies the set of connection coordinates of both structures. In this equation, [M] is the boolean mapping matrix which is used to establish the connectivity between the two substructures for synthesis. The mapping matrix is determined by the connectivity i.e. what is connected to what and by imposing the equilibrium and compatibility relations associated with each pair of coordinates. We can define the mapping matrix by $\{f_c\} = [M]\{\tilde{f}_c\}$. Where $\{f_c\}$ is a vector of all the connection coordinates of both structures and $\{\tilde{f}_c\}$ is the arbitrarily selected independent subset of the connection coordinates relating to one of the substructures. We have selected the connection coordinates of structure 1 as the arbitrary subset of connection coordinates. The mapping matrix [M] is a matrix of size (6 x 3) and is depicted as:

$$[M] = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ -1 & -1 & -1 & -1 & -1 & -1 \\ & & & -1 & & \\ & & & & -1 & \end{bmatrix}$$

The FRF matrix [H] for both substructures 1 and 2 is calculated using the matrix modal representation relation discussed earlier. The important point here is that we are not using all the coordinates. The coordinates used from substructure 1 are 2, 4, 5, and 6. the first coordinate is the coordinate of interest and the rest are connection coordinates which must be used. The coordinates used from substructure 2 are just the required connection coordinates 1, 2, and 3. All six mode shapes for each coordinate are kept for the calculation of the FRF matrix. The FRF matrix $[H_1]$ is calculated by using the appropriate partitioning of the modal matrix $[\Phi_1]$. The diagram of the relation is shown below showing the coordinates kept and the number of modes.

$$[H_1] = \begin{matrix} & \begin{matrix} 2 \\ 4 \\ 5 \\ 6 \end{matrix} & \\ \begin{matrix} 2 \\ 4 \\ 5 \\ 6 \end{matrix} & \begin{bmatrix} \Phi_1 \end{bmatrix} & \end{matrix} \begin{matrix} & \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} \frac{1}{\omega_r^2 - \Omega^2} \end{bmatrix} & \end{matrix} \begin{matrix} & \begin{matrix} 2 \\ 4 \\ 5 \\ 6 \end{matrix} & \\ & \begin{bmatrix} \Phi_1^T \end{bmatrix} & \end{matrix}$$

The size of $[H_1]$ is now a (4 x 4) matrix which contains all the necessary information. This also shows a significant computational advantage because the size has been reduced from a (6 x 6) to a (4 x 4) matrix which requires less computational time to manipulate the matrix. $[H_2]$ is generated in the same manner. We will use all six mode shapes for each coordinate kept of substructure 2. The required coordinates are the connection coordinates 1, 2, and 3. The diagram of the relation is shown below showing the coordinates kept and the number of modes.

$$[H_2] = \begin{matrix} & \begin{matrix} 6 & & 6 & & 1 & 2 & 3 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \left[\begin{array}{ccc|ccc} & & & & & \\ & & & & & \\ & & & & & \\ \Phi_2 & & & & & \\ & & & & & \\ & & & & & \end{array} \right] \end{matrix}$$

Now with h_1 and h_2 generated, the two substructures can be synthesized. Referring to the example-specific synthesis equation above, the matrices $[H_{2c}]$ and $[H_{cc}]$ are formed by combining h_1 and h_2 by appropriate partitioning. The partitioning is shown below.

$$[H_{2c}] = \begin{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ i_1 & \left[\begin{array}{cc} H_1(i_1, c_1) & [0] \end{array} \right] \end{matrix} \quad [H_{cc}] = \begin{matrix} & \begin{matrix} c_1 & c_2 \end{matrix} \\ c_1 & \left[\begin{array}{cc} H_1(c_1, c_1) & [0] \\ [0] & H_2(c_2, c_2) \end{array} \right] \end{matrix}$$

Referring to Figure 8.2, "i1" is the set of internal coordinates for substructure 1. Since coordinate 2 is the only coordinate of interest, the set of internal coordinates is just coordinate 2. The set of connection coordinates "c1" consists of 3, 4, and 5 and "c2" consists of 1, 2, and 3. With the appropriate partitioning complete, the two structures are synthesized together, to form the structure in Figure 8.1, using the case specific form of the direct coupling relation

$$[H_{2c}]^* = [H_{2c}] - [H_{2c}][M]([M]^T[H_{cc}][M])^{-1}[M]^T[H_{cc}].$$

$[H_{2c}]^*$ is the synthesized FRF by modal representation relation which is the combination of both structures. The synthesis is done over the frequency range of interest and plotted in Figure 8.3. The frequency range for this example was 0.1 to 80.0 Hz. Figure 8.4 is the solution from standard FE calculations using the inverse of the impedance matrix to calculate the FRF and Figure 8.5 is the solution from standard FE calculations using the modal representation to calculate the FRF. Figures 8.4 and 8.5 are included for direct comparison. All three plots are identical.

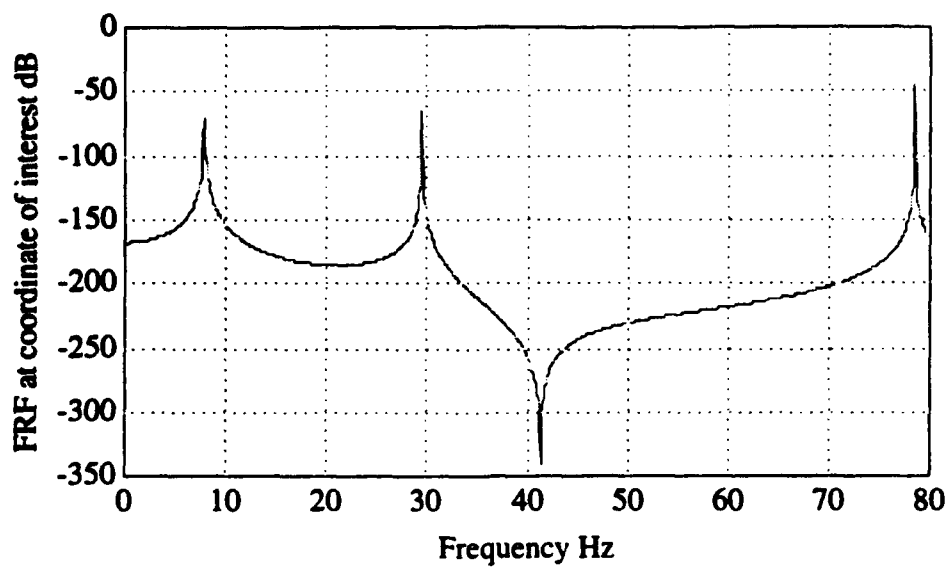


Figure 8.3 Plot of synthesized $H_{2c}(1,3)$

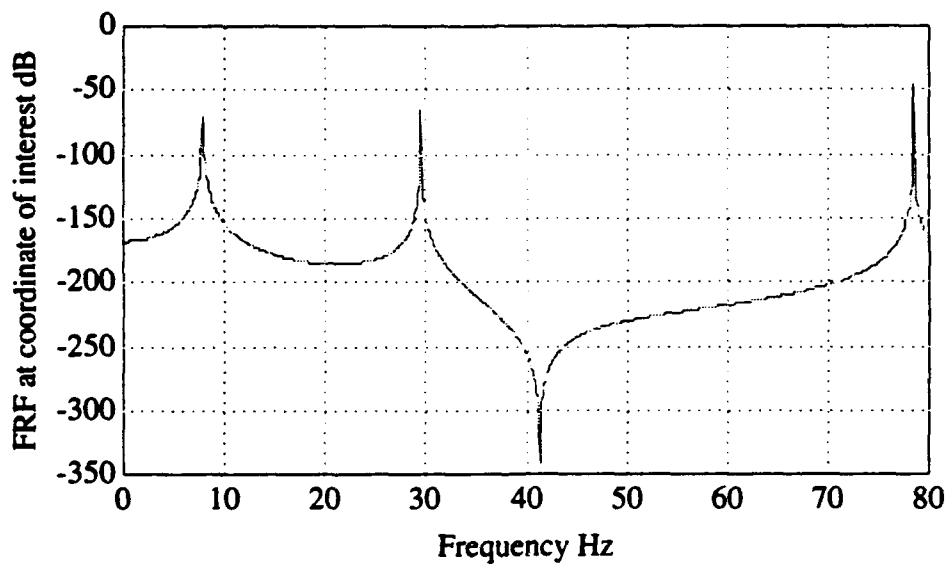


Figure 8.4 Plot of $H(2,6)$ from standard FE calculations

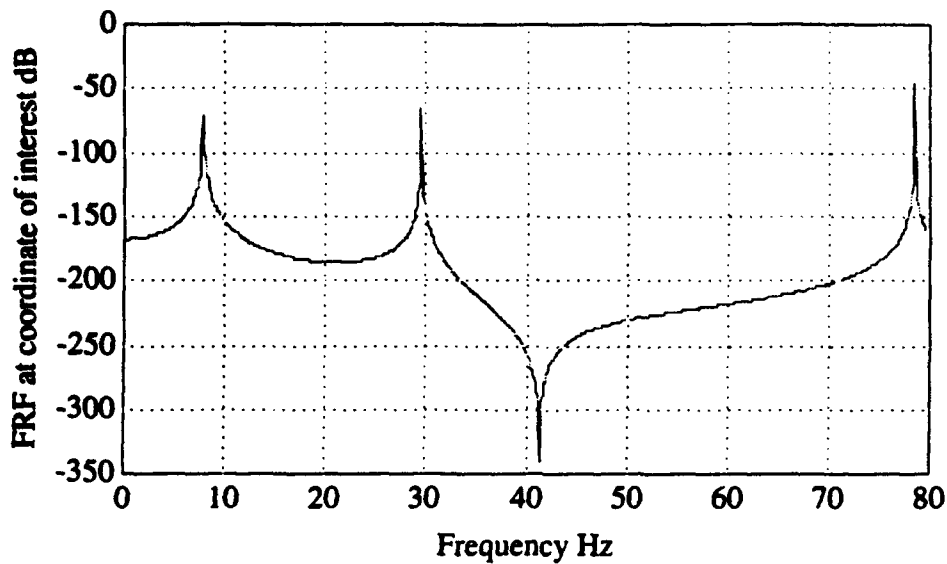


Figure 8.5 Plot of $H(1,4)$ from standard FE calculations using modal representation

Figures 8.3, 8.4 and 8.5 are the plots of the FRF at element (1,3), (2,6), and (1,4) respectively. These elements corresponds to the lateral motion coordinate 2 of Figure 8.1. A special note here is that the element (2,6) of the FRF generated by FEM is the response at coordinate 2, which corresponds to the element (1,3) of the synthesized FRF generated by the direct coupling relation using modal representation, and element (1,4) of the FRF generated by standard FE using modal representation. The reason for this is because of the partitioning and the coordinates used in the calculation. Care is required here to ensure the coordinate of interest is actually being used. The plots show the magnitude of the response at coordinate 2 due to a unit excitation at varying frequency at coordinate 6. As the frequency of excitation approaches the natural frequency of response, the response approaches infinity.

Figure 8.6 is the plot of the determinant of \tilde{H}_{cc} which shows the natural frequencies of the synthesized structure. The frequencies where the plot crosses the axis or equivalently, the frequencies for which the $\det[\tilde{H}_{cc}] = 0$ correspond to the natural frequencies of the synthesized structure. This information is important because it gives the designer a starting point on deciding how many modes to keep in the modeling of the system and the frequency bandwidth over which to perform the synthesis. Reducing the number of retained modes will decrease the computational cost and the computer time required to analyze a given design. The number of modes required to accurately model a given structure is case specific.

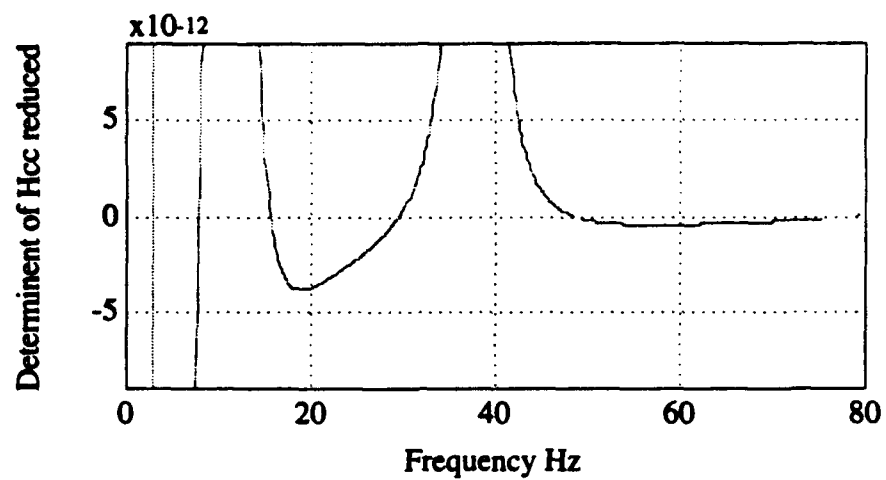


Figure 8.6 Plot of the determinant of \tilde{H}_{cc} (plotted over reduced bandwidth)

REFERENCES

- [1] Gordis, J. H., Bielawa, R. L., Flannelly, W. G., "A General Theory for Frequency Domain Structural Synthesis," *Journal of Sound and Vibration*, Vol. 150 No. 1. pp. 139-158. 1989.
- [2] Gordis, J. H., Flannelly, W.G. "Analysis of Stress Due to Fastener Tolerance in Assembled Components." *Proceedings of the 34th AIAA/ASME/ASCE/AHS/ACS Structures, Structural Dynamics, and Materials Conference*. La Jolla, CA. 1993.
- [3] Gordis, J. H. 1993. "Structural Synthesis in the Frequency Domain: A General Formulation." To appear, *Proceedings of the 12th International Modal Analysis Conference*, Honolulu, HI.

PART II

EXPERT SYSTEM STRUCTURE USING THE RATIONAL DESIGN SYSTEM (RDS) CONCEPT

1.0 ARTIFICIAL INTELLIGENCE AND RULE BASED SYSTEM CONCEPTS

1.1 General Overview

From one point of view, the field of Artificial Intelligence is associated with the use of computers to give human-like capabilities (such as learning, reasoning among alternatives, decision making, and self correction) to machines. In the context of this part of the report, we view the use of AI methodology only in a very limited sense, and as an aid to making decisions based on an explicit set of rules that structural designers would normally employ as a design is optimized. Design optimization must balance multiple checks all the way from static integrity under a variety of loading conditions, through shock isolation performance, and noise transmissibility to issues of cost and manufacturability.

Reasoning among a set of rules and providing suggestions to a human designer as to what to do next can certainly remove some of the laborious tasks required when searching over many alternatives. Computer based systems that perform such suggestions are called 'Knowledge Based Systems', and in particular, if the rules are compiled from resident human knowledge, they are called 'Expert Systems'. Expert Systems derive their name from the notion that they can automate the search through a set of rules representing the 'Knowledge Base' employed by the human expert and decide whether an operator entered query or question has a true or false answer. This notion can be extended to automating a search of a rule base that embodies decisions about what design parameters must be changed in arriving at a new design with better performance than a previous one.

The underlying concepts upon which computer based decisions are made lie in the use of an inference engine to search a given rule base where the rules are represented in terms of conditional statements involving combinatorial logic being associated with some action. So, in this report, we will firstly review the options for designing a rule base, different kinds of search methods, the use of AND/OR logic, state diagrams, and the use of 'PROLOG' as an instance of logic programming.

1.2 Artificial Intelligence, Logic, Rules and Rule Searches

The encoding of a set of rules that are normally used by experts to design a structural system differs from the common engineering computer code in its extensive the use of symbolic operations rather than numerical operations. We therefore use variables that are symbolic entities, associated with the attributes of TRUE or FALSE rather than variables that carry a particular

down', or backward reasoning are applied to this type of inference. Alternatively, deductive reasoning applies to the situation where a particular set of data or facts are taken and matched with conditions that are tied to rules so that subsequent conclusions can be deduced. This type of reasoning is sometimes called '*bottom up*' or '*data driven*' or '*forward*' reasoning.

1.3 Backward and Forward Chaining

To review the differences between forward and backward chaining we may return to the logical statements above. A forward chaining rule such as,

$$A, B \rightarrow C$$

is a representation of the rule that the condition (A and B is true) implies that C is true. Rules written in this fashion are suited for forward chaining solutions because the left-hand side A and B must be first evaluated from checking the current state of A and B from which a forward progression to the result that C must be true will follow. In a forward chaining system, the facts are reviewed and if left-hand sides of any rule is found to be true, that particular rule will be activated or 'fired' with the subsequent action that 'C is true. The new fact is added into the memory so that subsequent checking of the data base will also include the conclusion that 'C is true'. One drawback to this method of inference is that priority of multiple solutions can not be forced. Since choice among alternative solutions is a particular requirement for optimization of design, this method of rule search leaves something to be desired.

In a backward chaining search of the rules, a root or top goal is defined that is linked to sub goals through logical combinations using AND, OR and each sub-goal is further subdivided into sub-sub-goals. A search of the subgoal linkages is made until A BOTTOM GOAL is reached that is known to be true or false. Continued searching is done until the sub-goals can be proven true or false and continued until the root goal is proven.

Rules suited to the backward reasoning are written ideally as

$$C :- A, B$$

meaning C is true if A and B is true.

In backward reasoning, alternatives may be specified by the logical OR operation. For instance,

values 0 or 1. Thus the rule defines a structure for logically combining variables and Boolean algebra provides a means for the evaluation of variable combination. Rule search strategies relate to the ordering with which rules are traversed in solving the system of rules for the 'state', (0 or 1), of the total system of variables. Such a system of logical variables having values 0 or 1, linked by a set of rules, is called a finite state machine.

Predicate Calculus goes one step further because there is an inherent limitation in using simple logical variables to represent knowledge, and we often prefer to define logical functions which may include arguments as a predicate. In other words, a variable may contain one or more arguments. For example the logical variable:

select_mount_type(stiffness, gap)

can be considered as a predicate that is true if 'stiffness' and 'gap' have values taken from a particular defined set. If not, *select_mount_type (stiffness, gap)* would be considered false.

In this case, if we wish to make a rule statement equivalent to " select a high stiffness mount if shock induced equipment motions are too large ", we could write an example rule :

" *select_mount_type (high,small)* is true if *shock_motions(large)* is true"

in which the predicates 'stiffness' and 'gap' may take on the values 'high', 'moderate', 'low' or 'small', 'moderate', 'large', respectively.

Collecting a set of rules, however, that represent a knowledge base is not useful unless conclusions or resolutions can be reached from a 'reasoning' about the rule base as applied to a current case under study. It therefore follows that a mechanism called an 'inference engine' must be employed to search the rule base using some strategy, and present true or false answers to cases posed by the human operator. An inference engine searches the rules to determine what is known and to deduce from that, further facts that eventually lead to conclusions about what is to be found. The search mechanics are based on a matching of facts to rules in the knowledge base. A chaining of decisions follow that in general can be progressed in a forward or a backward fashion. Forward or Backward reasoning in logic represents the two distinctly different methods of reasoning - inductive and deductive. Inductive reasoning starts with a premise or goal and attempts to reason among facts that are linked to that goal. , Either a depth first - breath, or a breath first - depth search pattern can be employed for a search pattern. Sometimes the word goal decomposition, 'top

numerical value. An example could be, for instance, 'check_static_loads_ok' which could be either true or false. Decisions made on the basis of symbolic variables being 'T' or 'F' require a calculus for their evaluation and this is found in what is known as propositional logic and predicate calculus.

Propositional logic contains the basis for combining expressions that are true or false to yield others whose truth can be evaluated using the six primary logical operators AND, OR, NOT, IMPLIES (THEN), IF, AND ONLY IF. Decisions are made on the basis of *comparison* followed by an *action* and these take the form of a 'rule'

IF (*condition*), THEN (*action*)

For instance if A and B are symbolic variables that may be combined logically and C is another symbolic variable one example rule would be,

IF (A AND B is true), THEN (C is true).

In a mathematical sense this rule can be represented in either of two ways,

$$A, B \rightarrow C$$

i.e.,

'(A and B) is true' implies that 'C' is true

where the string A, B separated by a comma means the logical AND operation on A and B,

Alternatively, the same rule can be restated as an alternative to be discussed in more detail later as,

$$C : \sim A, B.$$

which means that

'C' is true if '(A and B)' is true.

Propositional logic deals with a calculus relating to the evaluation of logically combined variables. Boolean algebra is used for this purpose where the expressions A, B, take on the binary

$C :- A, B;$

$C :- D, E, F.$

will mean that

'C' is true if '(A and B)' is true, OR 'C' is true if '(D and E and F)' is true.

This ordering, with the express priority of first rule encountered having the highest priority, allows one to order the search of the rules to find the most desirable solution first but allowing alternative solutions to be found as possibly less desirable answers that may also be considered. In this case A and B being true is the preferred way to find C to be true, but D and E and F to be true is an acceptable alternative solution for C to be true.

A diagram of the AND / OR relationship is sometimes convenient to use in visualization of the logical flow and is given in Figure 1 where the natural progress of evaluating expressions is taken to go from left to right, and the left hand apex denotes the logical AND combination of A and B, while the second grouping represents the logical AND combination of the statements D,E,F. In this notation, the logical OR implied by the presence of the second grouping and the notion of searching from left to right also carries the important fundamental implication that the OR combination is a secondary way in which the root goal of C may be satisfied.

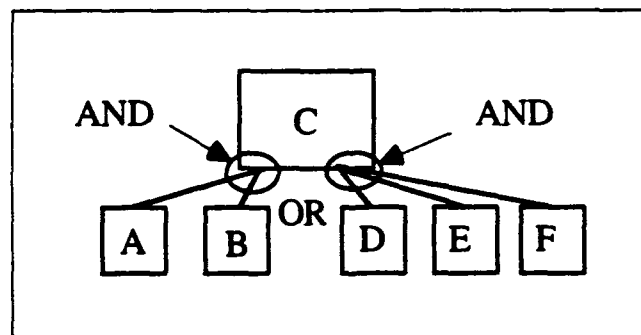


Figure 1 AND / OR Diagram As The Basis For The Rule Set

By way of a syntax, the secondary way of goal satisfaction through the logical OR function, represents an ability to build in priorities among rules as the inference engine searches the rule base from left to right to answer queries.

1.4 The Tri-Level Architecture Concept

Intelligent systems require an architecture to best structure their design and operation. In this sense, the combination of logical and symbolic computation with hard numerical processing required by both robotic systems as well as design of mechanical systems is often enabled by a tri-level architecture. By analogy to robotic systems, the three levels include the STRATEGIC level of design which determines basically WHAT to do next as the top level, the TACTICAL level in the middle which basically determines HOW to do the next step, where choices among alternatives may be made, and the EXECUTION level at the bottom which performs the numerical processing required to evaluate hard numerical values for responses in particular cases. The special name of RATIONAL DESIGN SYSTEM is defined as the architecture of a system that simplifies the decisions at the top to retain no memory of the state of the design and retain only the logical state of the inference engine in determining what to recommend next while the middle TACTICAL level retains the working memory, decisions among alternatives and is the interface between the symbolic and the numerical processing which takes place at an EXECUTION level.

The concepts above have been developed over the last few years in connection with the intelligent control of mobile robotic vehicles including walking machines[1], and unmanned underwater vehicles [2,3]. Our thinking on software architectures for control of vehicles has been most recently encapsulated in the Ph.D. dissertation of Byrnes [4], who has clearly outlined the Rational Behavior Model (RBM) out of which this section has grown.

The characteristics of a RATIONAL DESIGN SYSTEM (RDS) are illustrated by the diagram in Figure 2 below.

STRATEGIC LEVEL
<ul style="list-style-type: none"> • Symbolic computation only; contains the design doctrine. • No storage of the state of the design parameters and variables. • Rule based implementation. • Directs the Tactical Level through messages that are queries as to the truth of the predicates.
TACTICAL LEVEL
<ul style="list-style-type: none"> • Memory and Design Data reside here. • Directs Execution Level through procedures and requests for computations. • Responds to Strategic Level queries by comparison of current state of the design with performance requirements. • Establishes what computations to do to answer queries from above.
EXECUTION LEVEL
<ul style="list-style-type: none"> • Provides the numerical computations required by the Tactical Level and sends back data corresponding to the current state of the design.

Figure 2 Characteristics of the R D S

2.0 POSSIBLE RULE BASES FOR CRADLE DESIGN

2.1 General Overview

Because the nature of mechanical system design naturally involves the selection of particular solutions among alternatives, the work of this study has focused on the use of *backward reasoning* and rules written in the *backward chaining* format typified by root goal definition, and top down goal decomposition and a depth first - breath search strategy for rule satisfaction.

2.2 Use of 'PROLOG' Predicates for Strategic Level Rules

The AI rule based computer language, 'PROLOG' for PROgramming in LOGic has been found to be extremely easy and useful as a natural language for the specification of robotic missions and is adapted here as a natural specification language for the design optimization process. 'PROLOG' rules for the strategic level are listed as an example for the root goal of

'design(done)'.

In order to complete a design of a machinery cradle structure discussion with personnel at General Dynamics, Electric Boat Division, December 1993, have indicated that a new design must be checked for static loading, shock loading, noise transmission, and general feasibility as far as cost is concerned. Static loading checks primarily involve the deflections in the structure and its mountings, checks for acoustic shorting, and checks for displacements of the connecting hardware such as attached piping. To represent the checking processes of design, we elect to say that the design is done if each check, performed in a particular sequence of priorities meets the associated constraints. If a particular check meets its constraints, we say that a PROLOG predicate describing that check is 'TRUE'. The design is done when all checks return the answer TRUE. If any check returns the answer FALSE, then the design is not 'done'. Logically this means that all the checks must be combined with the logical AND function and that if any check is FALSE, design(done) is automatically unsatisfied or false. The correct logic is stated by the PROLOG predicate rule;

```
design(done) :-      select_new_design, check_statics_ok, check_shock_ok, check_noise_ok,
                    check_cost, check_weight_ok, case_optimized.
design(done) :-      design_close(Z), Z=='ok'.
```

What this means is that the root goal of 'design(done)' is decomposed into the subgoals of

select_new_design,
check_statics_ok,
check_shock_ok
check_noise_ok
check_cost
check_weight_ok,
case_optimized

which are joined in an AND relationship in order for the root 'design(done)' to be TRUE.

Colloquially the rule states that 'design(done)' is true if 'select_new_design' AND 'check_statics_ok' AND 'check_shock_ok' AND 'check_noise_ok' AND 'check_cost' AND 'case_optimized' are true, OR 'design_close(Z)' AND 'Z='ok' is TRUE.

A single primary rule of this type is initially quite sufficient to see if the design is complete, however, because of a detail in the PROLOG code compiler and its interfacing with 'C' language procedures used to interface the logical predicate rules to the numerical computational procedures, we need to close off the activity of the design(done) queries by guaranteeing that design(done) will ultimately become complete in some form. The completion of the query for design(done) can then be guaranteed by a second level rule that will logically combine the predicate design_close(Z), and the answer ' Z='ok' ' as in,

design(done):- design_close(Z), Z='ok'.

Use of this approach leads to an easy statement of what has to be done in order for the design to be completed. Further decomposition now may occur by discovering what constitutes the subgoals of the above named subgoals. In other words, ' select_new_design ' could be completed a number of different ways, as shown in Figure 3 which details an initial view of the entire set of rules for the strategic level of the RDS.

```

design(done) :- open_case(A),select_new_design,check_statics_ok,check_shock_ok,
               check_noise_ok,check_cost,check_weight_ok,case_optimized,design_close(Z),Z='ok'.

design(done):- design_close(Z),Z='ok'.

select_new_design:-select_new_mount_type.
select_new_design:-select_new_mount_locations.
select_new_design:-select_new_equipment.
select_new_design:-select_new_equipment_locations.
select_new_design:-select_new_structure.

check_statics_ok:-check_level_loads,check_pitch,check_roll,check_temp,
                  check_contraction,check_operation_loads(L),L='ok'.

check_shock_ok:-run_shock(X),X='yes'.

check_noise_ok:-db_budget(Y),Y='ok'.
check_noise_ok:-run_noise(Z),Z='ok'.

check_cost:-run_cost(A),A='ok'.

check_weight_ok:-run_weight(B),B='ok'.

case_optimized:-case_accepted(C),C='yes'.

select_new_mount_type:-change_gap.
select_new_mount_type:-change_stiffness.

change_gap:-gap_increased.
change_gap:-gap_decreased.

gap_increased:-run_shock(D),D='no',gap_increase_possible(E),E='yes'.
gap_decreased:-check_gap(G),G = 'too_large',gap_decrease_possible(H),H='yes'.

change_stiffness:-stiffness_increase.
change_stiffness:-stiffness_decrease.

stiffness_increase:-run_shock(D),D='no',stiffness_increase_possible(K),K='yes'.

stiffness_decrease:-run_noise(N),N='no',stiffness_decrease_possible(L),L='yes'.
select_new_mount_locations:-change_number_mounts.
change_number_mounts:-1==2.
select_new_equipment:-1==2.
select_new_equipment_locations:-1==2.
select_new_structure.
check_level_loads:-run_level_loads(M),M='yes'.
check_pitch:-check_pitch_loads(P),P='ok'.
check_roll:-check_roll_loads(R),R='ok'.
check_temp.
check_contraction:-check_contraction_loads(T),T='ok'.

```

Figure 3 Rules for the Strategic Level of the RDS

In the case at hand, 'select_new_design' was considered to be TRUE when either

```
'select_new_mount_type'  
'select_new_mount_locations'  
'select_new_equipment'  
'select_new_equipment_locations'
```

were completed or the 'design' was closed and the reply Z='ok' is received. As we continue with the organization, we will see that a certain style of definition of the PROLOG rules should be used. It helps to be as clear as possible in the specification of predicates so that the natural language used by engineers can be expressed. This makes rule definition and understanding easy for non-specialists.

At this point, a discussion of the use of predicates that have integer, string or character arguments is needed. It is quite convenient to be able to link a predicate with arguments to a calculation procedure used to compute the answer for the argument. Thus hooks to a 'C' language procedure exist in the form of linkage code involving foreign files written in 'C' which can return argument values in virtually any form desired. In particular, returning a 'yes' or 'no' answer becomes useful and indeed necessary to link the rules to computational processes.

The predicate, 'select_new_mount_type', is true if a new mount is selected. We have built in ways to satisfy this goal through gap and stiffness changes, and the first priority given to gap changing as in;

```
select_new_mount_type :- change_gap.  
select_new_mount_type :- change_stiffness.
```

and the decomposition into,

```
change_gap :- gap_increased.  
change_gap :- gap_decreased.
```

further decomposed into,

gap_increased :- run_shock(D), D='no', gap_increase_possible(E), E='yes'.

gap_decreased :- check_gap(G), G='too_large', gap_decrease_possible(H), H='yes'.

Clearly in the case of the predicate 'gap_increase_possible(E)' there will be a linkage to an equivalent C procedure that will reside in the TACTICAL level having access to the global memory of the state of the design, the performance requirements, and the limiting values of the design parameters, so that a 'yes' 'no' answer to the query 'is an increase in gap possible?' may be computed.

The rule for 'change_stiffness' is similarly decomposed into a stiffness increase or decrease, as in,

change_stiffness :- stiffness_increase.

change_stiffness :- stiffness_decrease.

with

stiffness_increase :- run_shock(D), D='no', stiffness_increase_possible(K), K='yes'.

stiffness_decrease :- run_noise(N), N='no', stiffness_decrease_possible(L), L='yes'.

The second predicate in the root goal check is 'check_statics_ok'. Clearly this is intended to require a complete checking of the static loading against criteria established and embodied in the data banks of the TACTICAL level. So far in this project, we have decomposed the statics checks into predicates that represent loading from pitch and roll loadings, level loadings, temperature loadings, contraction loadings as the submarine goes deep, and operational loadings from normal running of the equipment that would be carried by the cradle structure. It follows that the sub goal 'check_statics_ok', may be decomposed into,

check_statics_ok :- check_level_loads, check_pitch, check_roll, check_temp, check_contraction,
check_operational_loads(L), L='ok'.

and the further decomposition into,

check_level_loads :- run_level_loads(M), M='ok'.

check_pitch :- run_pitch_loads(P), P='ok'.

check_roll :- run_roll_loads(R), R='ok'.

`check_contraction :- run_contraction_loads(T), T='ok'.`

`check_temp :- run_temp_loads(Q), Q='ok'.`

Further decomposition of the subgoals associated with the root goal follow and an example of a STRATEGIC level code using the ideas outlined has been developed together with the foreign file hooks to the 'C' language procedures and run to verify its validity.

2.4 PROLOG / C Interfacing

It will be observed that the strategic level of RDS has to query the current state of the design for instance, to find out if it is possible to reduce the mount gap. This query, is made through a variable which carries an argument. The argument obtains its value from a C procedure resident in the TACTICAL level. Hence, there must be linkages between PROLOG and C through PROLOG Foreign files. The linkages are made by interfaces on the PROLOG side as well as C procedure links on the C side. Code has been developed for both sides of the linkage and is given in Figure 4 and the subsequent pages.

```

foreign_file(c,[c1,open_case,c11,c21,c3,c4,c5,run_shock,
check_operation_loads,
db_budget,
run_noise,
run_cost,
run_weight,
case_accepted,
check_gap,
run_level_loads,
check_pitch_loads,
check_roll_loads,
check_contraction_loads,
design_close,
gap_increase_possible,
gap_decrease_possible,
stiffness_increase_possible,
stiffness_decrease_possible
])).

foreign(c1, c, c1(+integer,[-integer])).
foreign(open_case, c, open_case(-integer)).
foreign(c11, c, c11(+atom, [-atom])).
foreign(c21, c, c21(+atom, -atom)).
foreign(c3, c, c3(+float, [-float])).
foreign(c4, c, c4(-float)).
foreign(c5, c, c5(+string,[-string])).
foreign(check_operation_loads, c, check_operation_loads(-string)).
foreign(db_budget, c, db_budget(-string)).
foreign(run_noise, c, run_noise(-string)).
foreign(run_cost, c, run_cost(-string)).
foreign(run_weight, c, run_weight(-string)).
foreign(case_accepted, c, case_accepted(-string)).
foreign(check_gap, c, check_gap(-string)).
foreign(stiffness_increase_possible, c, stiffness_increase_possible(-string)).
foreign(run_level_loads, c, run_level_loads(-string)).
foreign(check_pitch_loads, c, check_pitch_loads(-string)).

foreign(check_roll_loads, c, check_roll_loads(-string)).
foreign(check_contraction_loads, c, check_contraction_loads(-string)).
foreign(run_shock, c, run_shock(-string)).

foreign(design_close, c, design_close(-string)).
foreign(stiffness_decrease_possible,c,stiffness_decrease_possible(-string)).
foreign(gap_increase_possible,c,gap_increase_possible(-string)).
foreign(gap_decrease_possible,c,gap_decrease_possible(-string)).

```

Figure 4 PROLOG Side Foreign Files

The next few pages list the C side Links

C Side linkages to PROLOG Expressed as "C" VOID FUNCTIONS

```
#include <stdio.h>
FILE *temp;

/* c1(+integer, [-integer]) */

long int c1(a)
long int a;
{
    return(a+9);
}

/* open_case(-integer) */

void open_case(a)
long int *a;
{
    temp=fopen("logfile","a");

    *a = 99;
}

/* c11(+atom [-atom]) */

unsigned long c11(a)
unsigned long a;
{
    return(a);
}

/* c21(atom, -atom) */
void c21(a,b)
unsigned long a;
unsigned long *b;
{
    *b = a;
}

/* c3(float, [-float]) */

double c3(a)
double a;
{
    return(a+9.0);
}

/* c4(-float) */

void c4(a)
float *a;
{
    *a = 9.9;
}

/* c5(string, [-string]) */
char * c5(a)
char * a;
{
    return(a);
}

/* run_shock(-string) */

void run_shock(a)
```

```

char * *a;
{
    char test;

    fprintf(temp, "\n checking for shock \n");

    *a="yes";
    test='y';
    if (test=='y')
        {fprintf(temp, "\n shock motions are within limits \n");}
    if (test=='n')
        {fprintf(temp, "\n shock motions are too large \n");}
}

```

```

void check_operation_loads(a)

```

```

char * *a;
{

```

```

    char test;
    /* temp=fopen("logfile","a"); */
    fprintf(temp, " \n checking for operational loads \n");

    test='y';

    if (test=='y')
        {fprintf(temp, "\n op loads are ok\n");}
    if (test=='n')
        {fprintf(temp, "\n op loads are not ok \n");}

```

```

        *a="ok";
    /*      *a="not_ok";      */
}

```

```

void db_budget(a)

```

```

char * *a;
{

```

```

    char test;
    /* temp=fopen("logfile","a"); */
    fprintf(temp, " \n checking for db budget \n");

    test='y';

    if (test=='y')
        {fprintf(temp, "\n db has been accepted\n");}
    if (test=='n')
        {fprintf(temp, "\n db has not been accepted\n");}

```

```

        *a="ok";
    /*      *a="not_ok";      */
}

```

```

void run_noise(a)

```

```

char * *a;
{

```

```

    char test;
    /* temp=fopen("logfile","a"); */
    fprintf(temp, " \n checking for noise \n");

    test='y';

    if (test=='y')
        {fprintf(temp, "\n noise has been accepted\n");}
    if (test=='n')

```

```
{fprintf(temp, "\n noise has not been accepted\n");}
```

```
    *a="ok";  
/*    *a="not_ok";    */  
}
```

```
void run_cost(a)
```

```
char * *a;
```

```
{  
    *a="ok";  
/*    *a="not_ok";    */  
}
```

```
void run_weight(a)
```

```
char * *a;
```

```
{  
    *a="ok";  
/*    *a="not_ok";    */  
}
```

```
void case_accepted(a)
```

```
char * *a;
```

```
{  
char test;  
    /*temp=fopen("logfile","a");*/  
    fprintf(temp, " \n checking for case_accepted \n");  
  
    *a="yes";  
    test='y';  
  
    if (test=='y')  
        {fprintf(temp, "\n case has been accepted\n");}  
    if (test=='n')  
        {fprintf(temp, "\n case has not been accepted\n");}  
}
```

```
void check_gap(b)
```

```
char * *b;
```

```
{  
char test;  
    /*temp=fopen("logfile","a");*/  
  
    fprintf(temp, "\n checking for change_gap \n");  
  
    *b="too_large";  
    test='n';  
  
    if (test=='y')  
        {fprintf(temp, "\n gap is acceptable\n");}  
    if (test=='n')  
        {fprintf(temp, "\n gap is too large\n");}  
}
```

```
void stiffness_increase_possible(b)
```

```
char * *b;
```

```
{  
    char test;  
/*    temp=fopen("logfile","a");  
*/  
    fprintf(temp, "\n checking for change_stiffness \n");  
  
    *b="yes";
```

```

    test='y';
    if (test=='y')
        {fprintf(temp, "\n stiffness increase is possible \n and has been increased\n");}
    if (test=='n')
        {fprintf(temp, "\n stiffness has not been changed\n");}

```

```

}
void run_level_loads(a)
char * *a;

```

```

{

char yes, no, test;
yes='y';
no='n';

/*      temp=fopen("logfile","a");
*/
    fprintf(temp, "\n checking level loads\n");

    *a="yes";
    test='y';

    if (test==yes)
    {
        fprintf(temp, "\n level loads check is ok\n");
    }
    if (test==no)
    {
        fprintf(temp, "\n level loads check is not ok\n");
    }
}

```

```

void check_pitch_loads(a)
char * *a;
{

```

```

char test;
/*      temp=fopen("logfile","a");
*/
    fprintf(temp, "\n checking pitch loads\n");

    *a="ok";
    test='y';
/*      *a='not_ok';*/

    if (test=='y')
        {fprintf(temp, "\n pitch loads check is ok\n");}
    if (test=='n')
        {fprintf(temp, "\n pitch loads check is not ok\n");}

}

```

```

void check_roll_loads(a)
char * *a;
{

```

```

char test;
/*      temp=fopen("logfile","a");
*/
    fprintf(temp, "\n checking roll loads\n");

```

```

        *a="ok";
        test='y';
        if (test=='y')
            {fprintf(temp, "\n roll loads check is ok\n");
            }

        if (test=='n')
            {
                fprintf(temp, "\n roll loads check is not ok\n");
            }
    }

void check_contraction_loads(a)
char * *a;

{

char test;
/*      temp=fopen("logfile","a");
*/

    fprintf(temp, "\n  checking contraction loads\n");

    *a="ok";
    test='y';
    if (test=='y')
        {
            fprintf(temp, "\n contraction loads check is ok\n");
        }
    if (test=='n')
        {
            fprintf(temp, "\n contraction loads check is not ok\n");
        }
    }

void design_close(a)
char * *a;
{
    fclose(temp);
    *a="ok";
}

void stiffness_decrease_possible(a)
char * *a;
{
    char test;

/*      temp=fopen("logfile","a");
*/

    fprintf(temp, "\n  checking for change_stiffness \n");

    *a="yes";
    test='y';
    if (test=='y')
        {fprintf(temp, "\n stiffness decrease is possible \n and has been decreased\n");}
    if (test=='n')
        {fprintf(temp, "\n  stiffness has not been changed\n");
        }
    }
}

```



```

}

void gap_increase_possible(a)
char * *a;
{
    char test;

    /*
    temp=fopen("logfile","a");
    */
    fprintf(temp, "\n checking for gap increase possibility \n");

    *a="yes";
    test='n';
    if (test=='y')
        {fprintf(temp, "\n gap increase is possible \n and has been increased \n");}
    if (test=='n')
        {fprintf(temp, "\n gap has not been changed\n");}

}

```

```

void gap_decrease_possible(a)
char * *a;
{
    char test;
    /*
    temp=fopen("logfile","a");
    */

    fprintf(temp, "\n checking for gap decrease possibility \n");

    *a="yes";
    test='y';
    if (test=='y')
        {fprintf(temp, "\n gap decrease is possible and has been decreased\n");}
    if (test=='n')
        {
            fprintf(temp, "\n gap has not been changed\n");
        }

}

```

3.0 EXAMPLE STRATEGIC LEVEL DESIGN

3.1 Introduction

In this section we present examples of the logical traces generated from a logfile and a debugging routine that help to establish the flow of the inference engine as rules are traversed. First a perfect design case where all checks are satisfied is run followed by a case in which various checks fail to pass so that the logic in changing the design may be followed to verify the correctness of the rules.

3.2 Logical Traces - Satisfactory Design Case

The logical trace of the satisfactory design in which all checks pass is given in the Figure 5 below

```
checking for shock
shock motions are within limits
checking for change_gap
gap is too large
checking for gap decrease possibility
gap decrease is possible and has been decreased
checking level loads
level loads check is ok
checking pitch loads
pitch loads check is ok
checking roll loads
roll loads check is ok
checking contraction loads
contraction loads check is ok
checking for operational loads
op loads are ok
checking for shock
shock motions are within limits
checking for db budget
db has been accepted
checking for case_accepted
case has been accepted
```

Figure 5 Logical Trace of Rules - All Checks Pass

3.3 Logical Traces - Failed Analyses

The logical trace from a case where the shock loading check failed is given in the Figure 6 below.

```
checking for shock
shock motions are too large
  checking for gap increase possibility
gap has not been changed
  checking for change_gap
gap is too large
  checking for gap decrease possibility
negative reply, gap has not been changed
  checking for shock
shock motions are too large
  checking for change_stiffness
stiffness increase is possible
and has been increased
  checking level loads
level loads check is ok
  checking pitch loads
pitch loads check is ok
  checking roll loads
roll loads check is ok
  checking contraction loads
contraction loads check is ok
  checking for operational loads
op loads are ok
  checking for shock
shock motions are too large
  checking for noise
noise has been accepted
  checking level loads
level loads check is ok
  checking pitch loads
pitch loads check is ok
  checking roll loads
roll loads check is ok
  checking contraction loads
contraction loads check is ok
  checking for operational loads
op loads are ok
  checking for shock
shock motions are too large
```

Figure 6 Logfile for the Case where the Shock Loading Design Check Fails as Motions are Too Large

CONCLUSIONS

What the results have shown so far is that it is definitely possible to develop a set of rules that would drive a design to a solution using engineering rules. But what has not yet been shown is that the solution obtained is optimized by the rules chosen. This crucial next step needs to be explored in detail and could be readily accomplished by linking the "C" code hooks to the PROLOG rules to additional "C" procedures that represent the world data base and the execution level coding that run the static and dynamic checks needed. In essence, we need to build the TACTICAL and EXECUTION levels of the RDS in order to show how easily the rules can be modified to provide optimization of the design.

REFERENCES

- [1] Kwak, S. H. McGhee, R. B. "Rule Based Coordination for a Hexapod Walking Machine," *Advanced Robotics*, Vol. 4, No. 3, pp 263-282 (1990).
- [2] Byrnes, R. B., MacPherson, D.L., Kwak, S.H., McGhee, R. B., Nelson, M.L., "An Experimental Comparison of Hierarchical and Subsumption Software Architectures for Control of an Autonomous Underwater Vehicle" *Proceedings of IEEE Oceanic Engineering Society Symposium on Autonomous Underwater Vehicles*, AUV-92 Washington D.C., June 2-3, 1992. pp.135-141
- [3] Byrnes, R. B., Kwak, S. H., McGhee, R. B., Healey, A. J., Nelson, M.,L., " The Rational Behavior Model - An Implemented Tri Level Multilingual, Software Architecture for Control Of Autonomous Vehicles ", *Proceedings of the 8th. UUST Symposium*, Durham New Hampshire Oct. 27-28 1993 pp. 160-179.
- [4] Byrnes, R. B., "The Rational Behavior Model: A Multi-Paragigm Tri-Level Software Architecture For The Control Of Autonomous Vehicles," Ph.D. Dissertation, Naval Postgraduate School, Monterey, CA. MARCH 1993.

INITIAL DISTRIBUTION LIST

	<u>No. of Copies</u>
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2. Mr. Dan Dozier Project Manager Program Executive Office - Submarines PEO-SUB-RC4, NC3 RM 6W52 2531 Jefferson Davis Highway Arlington, VA 22242-5160	1
3. Dr. Art Spero Program Manager, Submarine H M & E Code PEO-SUB-RB Naval Sea Systems Command Washington, DC 20362	1
4. Mr. Tom Berry General Dynamics - Electric Boat Division Groton, CT	1
5. Professor J. H. Gordis, Code ME/Go Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93943	4
6. Professor A. J. Healey, Code ME/Hy Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93943	3
7. LT Ronald E. Cook 1090 Derby St. Casper, WY 82609	1
8. Naval Engineering Curricular Office, Code 34 Naval Postgraduate School Monterey, California 93943-5000	1
9. Library, Code 52 Naval Postgraduate School Monterey, California 93943-5002	1